

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.

[19]中华人民共和国国家知识产权局

[51]Int. Cl⁶

HD4L 12/40

[12] 发明专利申请公开说明书

[21] 申请号 98800788.6

[43]公开日 1999年9月22日

[11]公开号 CN 1229552A

[22]申请日 98.4.30 [21]申请号 98800788.6

[30]优先权

[32]97.4.30 [33]JP [31]126355/97

[86]国际申请 PCT/JP98/01995 98.4.30

[87]国际公布 WO98/49808 日 98.11.5

[85]进入国家阶段日期 99.2.9

[71]申请人 索尼公司

地址 日本东京都

[72]发明人 中村章 大川纯弘 泷塚博志

[74]专利代理机构 柳沈知识产权律师事务所

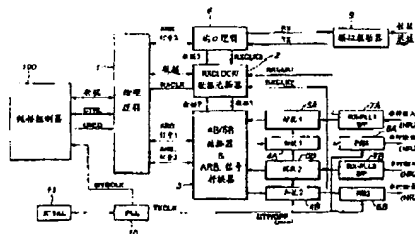
代理人 吕晓章

权利要求书 5 页 说明书 26 页 附图页数 15 页

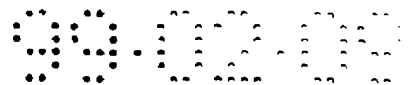
[54]发明名称 数字数据传输装置及方法

[57]摘要

一种用于数字数据传输的器件,它提供了转换块(4B/5B CONVERTER AND ARB SIGNALCONVERTE)(3),其不仅起到进行4位/5位数据转换的4位/5位转换装置的作用,而且起到给仲裁信号分配不同于4位/5位转换过程中分配给数数的5位码元的仲裁信号转换装置,因此该装置能够以5位码数据传输和接收仲裁信号和数据包。由于该器件在上面的转换块(3)和各个输入和输出端口之间提供了加扰块(SCRAMBLE1和SCRAMBLE 2)(4A和4B)和解扰块(DESCRAMBLE1和DESCRAMBLE 2),该器件能够在传输线上防止不必要的辐射。



ISSN 1008-4274



权 利 要 求 书

1. 一个发送和接收数据和控制代码的数据传输装置, 包括:
一个输入/输出端口;
- 5 数据转换装置, 用于把传输的数据从 n 位码转换成 m 位码并把接收的数据从 m 位码转换成 n 位码;
控制信号转换装置, 用于把获得与所述输入/输出端口相连的传输通道的使用权的传输控制信号转换成由不同于分配给数据的 m 位码的 m 位码构成的控制码以及用于把接收到的 m 位码转换成控制信号;
- 10 加扰装置, 用于对从所述控制信号转换装置接收到的 m 位控制码进行加扰以减少不必要的辐射以便将加扰的控制码输出给所述的输入/输出端口; 和
解扰装置, 用于对通过所述输入/输出端口接收到的加扰信号解扰以便将解扰的信号输出给所述的控制信号转换装置。
2. 根据权利要求 1 的数据传输装置, 其中所述的解扰装置检测在非数据传输时间发送的空闲信号以便随后实施解扰的同步。
- 15 3. 根据权利要求 2 的数据传输装置, 其中所述的解扰装置在结尾处数次检测到在非数据传输时间发送的空闲信号后接着实施解扰的同步。
4. 根据权利要求 1 的数据传输装置, 其中所述的空闲信号是有连续“1”的 m 位码。
- 20 5. 根据权利要求 1 的数据传输装置, 其中所述的解扰装置检测用于重置与所述输入/输出端口相连的传输通道的总线重置信号以便实施解扰同步。
6. 根据权利要求 5 的数据传输装置, 其中所述的解扰装置在结尾处数次检测到用于重置与所述输入/输出通道相连的传输通道的总线重置信号后接着实施解扰同步。
- 25 7. 根据权利要求 6 的数据传输装置, 其中所述的总线重置信号是有两个 m 位码的分配码元的控制码。
8. 根据权利要求 7 的数据传输装置其中所述的总线重置信号是一控制码, 该控制码由第一个具有连续“0”的 m 位码和第二个具有连续“1”的 m 位码构成。
- 30 9. 根据权利要求 1 的数据传输装置, 其中所述的加扰装置进行使 m 位控制码沿着时间轴方向扩散的处理。



10. 根据权利要求 1 的数据传输装置, 其中所述的加扰装置包括保持加扰的密钥格式(pattern)的密钥保持装置和用于对从所述密钥格式获得的密钥数据和从所述的控制信号转换装置输出的控制码得出异或的运算装置。

11. 根据权利要求 1 的数据传输装置, 其中所述的加扰装置包括对从所述的输入/输出端口接收到的输入加扰信号和所述的加扰输入信号的临时移位版本(version)执行预置的异或 OR 操作的虚拟(hypothesis)装置;

第一格式保持装置, 用于保持比较格式;

第一比较装置, 用于将所述的虚拟装置的输出数据与所述的第一格式保持装置的比较格式进行比较以输出相应于比较的结果的预置的密钥格式;

10 测试(tentative)装置, 用于对由来自所述第一比较装置的密钥格式得到的密钥数据和经所述输入/输出端口发送到那里的加扰信号进行预置异或 OR 运算, 以便输出 m 位的暂时控制码;

第二格式保持装置, 用于保持比较的解扰格式;

15 第二比较装置, 用于比较来自所述测试装置的测试控制码和来自所述测试装置的比较的解扰格式, 以输出相应于比较结果的预置密钥格式; 和

输出装置, 用于对来自所述的第二比较装置的密钥数据和从所述输入/输出端口发送到那里的加扰信号进行预置异或 OR 电路运算以输出 m 位的控制码。

12. 根据权利要求 1 的数据传输装置, 其中所述的解扰装置包括: 虚拟装置, 用于对通过所述的输入/输出端口接收到的输入加扰信号和所述的加扰输入信号的暂时移位版本进行异或运算;

格式保持装置, 用于保持比较格式;

比较装置, 用于对所述虚拟装置的输出数据与所述格式保持装置中的比较格式进行比较以响应于比较结果输出预置的密钥格式; 和

25 输出装置, 用于对由来自所述比较装置的密钥格式得到的密钥数据和通过所述输入/输出端口发送到那里的加扰信号进行预置的异或 OR 运算以输出 m 位的控制码。

13. 根据权利要求 1 的数据传输装置, 其中所述加扰装置和解扰装置的处理操作可以被打开或关掉。

30 14. 根据权利要求 1 的数据传输装置, 其中所述的数据转换装置将数据从 4 位码转换 5 位码。



15. 根据权利要求 1 的数据传输装置, 其中数据和控制信号从符合 IEEE 1394 高效串行总线标准的链接层进入。

16. 一数据传输方法, 包括:

发送数据转换步骤, 用于将传输的数据从 n 位码转换成 m 位码;

5 接收数据转换步骤, 用于将接收的数据从 m 位码转换成 n 位码;

传输控制信号转换步骤, 用于将获得与所述输入/输出端口相连的传输通道的使用权的传输控制信号转换成由不同于分配给数据的 m 位码的 m 位码构成的控制码及把接收到的 m 位码转换成控制信号;

10 加扰步骤, 用于为减少不必要的辐射对从所述的传输控制信号转换步骤接收到的 m 位控制码进行加扰以便将加扰的控制信号输出到输入/输出端口; 和

解扰步骤, 用于对通过所述输入/输出端口接收到的加扰信号进行解扰。

17. 根据权利要求 16 的数据传输方法, 其中所述的解扰步骤在检测到在非数据传输时间内发送的空闲信号以后接着实施解扰同步。

15 18. 根据权利要求 17 的数据传输方法, 其中所述的解扰步骤在结尾处多次检测到在非数据传输时间内传输的空闲信号以后接着实施解扰同步。

19. 根据权利要求 18 的数据传输方法, 其中所述的空闲信号是有连续“1”的 m 位码。

20 20. 根据权利要求 16 的数据传输方法, 其中所述的解扰步骤检测用于重置与所述输入/输出端口相连的传输通道的总线重置信号以实施解扰同步。

21. 根据权利要求 20 的数据传输方法, 其中所述的解扰步骤在结尾处多次检测到用于重置与所述输入/输出端口相连的传输通道的总线重置信号以后接着实施解扰同步。

25 22. 根据权利要求 21 的数据传输方法, 其中所述的总线重置信号是有两个 m 位码的分配码元的控制码。

23. 根据权利要求 22 的数据传输方法, 其中所述的总线重置信号由第一个有连续“0”的 m 位码和第二个有连续“1”的 m 位码构成的控制码。

24. 根据权利要求 16 的数据传输方法, 其中所述的加扰步骤使 m 位控制码进行沿时间轴方向扩散的处理。

30 25. 根据权利要求 16 的数据传输方法, 其中所述的加扰步骤包括密钥保持步骤, 用于保持加扰的密钥格式; 和计算步骤, 用于对由所述密钥格式获得

的密钥数据和由所述的控制信号转换步骤馈给的控制码进行异或运算。

26. 根据权利要求 16 的数据传输方法, 其中所述的加扰步骤包括: 虚拟步骤, 用于对通过输入/输出端口接收的输入加扰信号和所述的加扰输入信号的暂时移位版本执行预置的异或操作;

- 5 第一比较器步骤, 用于将所述虚拟步骤的输出数据与比较格式进行比较以输出相应于比较结果的预置密钥格式;

测试步骤, 用于对从所述的第一比较器步骤的密钥格式获得的密钥数据和通过输入/输出端口发送到那里的加扰信号进行预置的异或运算以输出 m 位的暂时控制码;

- 10 第二比较步骤, 用于对来自所述测试步骤的测试控制码和来自所述测试步骤的比较解扰格式进行比较以输出相应于比较结果的预置的密钥格式; 和

输出步骤, 用于对从所述的第二比较步骤的密钥格式获得的密钥数据和通过输入/输出端口发送到那里的加扰信号进行预置的异或 OR 电路运算以输出 m 位控制码。

- 15 27. 根据权利要求 16 的数据传输方法, 其中所述的解扰步骤包括: 虚拟步骤, 用于对通过输入/输出端口接收的输入加扰信号和所述的加扰的输入信号的暂时移位版本进行异或 OR 运算;

比较步骤, 用于将所述虚拟步骤的输出数据与比较格式进行比较以响应于比较结果输出预置的密钥格式; 和

- 20 输出步骤, 用于对从所述的比较步骤的密钥格式获得的密钥数据和通过输入/输出端口发送到那里的加扰信号进行异或 OR 运算以输出 m 位的控制码。

28. 根据权利要求 16 的数据传输方法, 其中所述的传输控制信号转换步骤把数据从 4 位码转换成 5 位码。

- 25 29. 根据权利要求 16 的数据传输方法, 其中数据和控制信号从符合 IEEE 1394 高效串行总线标准的链接层进入。

说明书

数字数据传输装置及方法

5 本发明涉及一种传送数字数据的方法及装置。

作为一种支持高速数据传输和实时传输的接口,为了数据传输的接口,就有了所知的 IEEE 1394 高效串行总线标准(IEEE 1394 标准)。

该 IEEE 1394 标准提供速率为 100Mbps(兆位/秒)(98.304Mbps), 200Mbps(196.608Mbps)和 400Mbps(393.216Mbps)的数据传输。具有较高级别
10 传输速率的 1394 端口,按照规定与其较低级别的传输速率保持兼容。这样, 100Mbps, 200Mbps 和 400Mbps 的数据传输速率可以在同一网络中共存。而且,在 IEEE 1394 标准中,使用 DS-LINK(DATA/STROBELINK)编码系统的传输格式,其中传输数据被转换成数据信号和补充数据信号的选通信号,如图 1 所示,其中的时钟信号是由两个信号取“异或”产生的。参照截面图
15 2 所示的电缆结构,规定了一种电缆 200,其结构为将两套由第一屏蔽层 201 屏蔽的双绞线(信号线)202 和电源线 203 捆扎并进一步由第二屏蔽层 204 屏蔽得到的电缆。

在 IEEE 1394 标准的连接系统中,可以使用两种系统,即菊花链系统和节点分支系统。使用菊花链,在节点最大距离为 4.5m 时最多有 16 个节点(一个有 1394 端口的设备)可以被连接。在联合中使用节点分支法,如图 3 所示,根据标准最多可以连结 63 个节点(物理节点地址)。

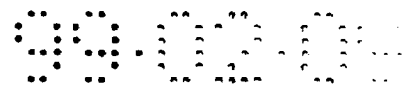
根据 IEEE 1394 标准,上述电缆的连结和断开可以在设备保持工作的状态下进行,即在加电源状态下,所以该 1394 网络能够在增加或减少节点的时
20 刻自动被重建。此时连接的节点的设备能够被自动地识别。可以在接口上管理连结设备的 ID 或布局。

遵从 IEEE 1394 标准的接口的组成元件以及协议结构如图 4 所示。IEEE 1394 的接口可以分为硬件和固件。

硬件由物理层(PHY)和被连结层(连结层)构成。

在物理层中, IEEE 1394 标准的信号直接被驱动。连结层具有主接口和
30 物理层。

固件包括一处理层,由一遵从 IEEE 1394 标准为接口进行实际操作的管理



理驱动器构成, 还有一管理层, 由一遵从 IEEE 1394 标准进行网络管理的驱动器构成, 称为串行总线管理(SBM)。

应用层包括一用户使用的软件和一用于连接管理层或处理层的管理软件。

5 在 IEEE 1394 标准中, 该网络进行的传输操作被称为子操作(sub-action), 对其规定了下面的两种子操作。这两种被规定的子操作即为, 一种异步传输模式, 称为“异步 (asynchronous)”, 和一提供传输区域的同步传输模式, 称为“等时(isochronous)”。每个子操作被分成三部分, 并假定传输状态称为“仲裁”, “包传输”和“确认”。

10 在异步子操作中使用异步传输。在图 5 中示意了该传输模式的暂时过渡状态, 第一子操作间隙规定了总线的空闲状态。通过监测子操作间隙时间, 直接前次的传输结束并给出新的传输是否可能的判定。

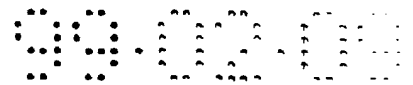
如果空闲状态持续的比预置的时间间隔长, 发出一个需要传输的节点可以使用总线的决定, 并且执行仲裁以获得总线的控控权。停止总线的决定实际上由位于根部的节点 B 发出, 如图 6a 和 6b 所示。通过此仲裁获得总线控制权的节点便开始数据传输, 即包传输。数据传输后, 收到数据的结点返送一代码询问(为确认接收而返送的代码)以响应收到的传输数据的结果。执行响应确认。通过执行此确认, 发送和接收的节点都可以通过询问的内容确定传输正常地结束了。

20 接着子操作间隙, 即总线闲状态被恢复以便重复上面的传输操作。

在等时子操作中, 执行一基本与异步传输相似的传输结构。如图 7 所示, 本传输的执行优先于异步子操作中的异步传输。本等时子操作中的等时传输的执行大约以每 8kHz 优先于异步子操作中的异步传输, 以设定一保证传输区的传输模式。这可以实现实时数据的传输。

25 如果要在多个节点中执行实时数据的等时传输, 在传输数据中设定一区分内容(传输节点)的通道 ID 以便只接收所需的实时数据。

上述 IEEE 1394 标准的物理层由一物理层逻辑块(PHY LOGIC)102, 一选择块(RXCLOCK/DATA SELECTOR)103, 端口逻辑块(PORT LOGIC 1, PORT LOGIC 2, PORT LOGIC 3)104, 105, 106, 电缆端口(CABLE PORT 1, CABLE PORT 2, CABLE PORT 3)107, 108, 109 和一时钟发生块(PLL)110 构成, 如图 8 所示。



物理层逻辑块 102 利用 IEEE 1394 标准中的连接层进行 I/O 控制和仲裁控制，它不仅与连接层控制器 100 而且与选择块 103 及端口逻辑块 104, 105, 16 相连接。

5 选择块 103 对从与电缆端口 107, 108, 109 相连的逻辑块 104, 105, 106 和接收时钟(RXCLK 1, RXCLK 2, RXCLK 3)收到的数据进行选择，它与物理层逻辑块 102 和端口逻辑块 104, 105, 106 连接。

接收时，每组从端口逻辑块 104, 105, 106 收到的数据包数据(DATA1, DATA2, DATA3)和接收时钟(RXCLK 1, RXCLK 2, RXCLK 3)被选择以便将收到的数据包数据和接收时钟经电缆端口 107, 108, 109 发送到物理层逻辑块 10 102。例如，如果选择了经电缆端口 107 再经端口逻辑块 104，收到的数据包 DATA 1 和接收时钟 RXCLK 1，接收到的数据包数据(DATA 1)和它的接收时钟 RXCLK 1 将由端口逻辑块 104 送到物理层逻辑块 102。经选择块 103 选择的数据包数据由接收时钟写入物理层逻辑块 102 的 FIFO 内存中。写入 FIFO 内存的数据包数据由时钟发生块 110 提供的系统时钟 SYSCLK 读出。

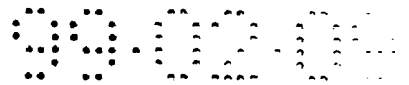
15 端口逻辑块 104 经过电缆端口 107 发送/接收仲裁信号(ARB.SIGNAL)和数据(DATA 1)，并具有从电缆端口 107 发送的数据信号和它的选通信号中产生接收时钟(RXCLK 1)的功能。在仲裁过程中端口逻辑块 104 由物理层逻辑块 102 馈给一仲裁信号(ARB.SIGNAL)。

在数据传输时间内，端口逻辑块 104 按时钟发生块 110 提供的传输时钟 20 TXCLK 将从物理层逻辑块 102 经选择块 103 发送的数据包 DATA 1 转换成通过电缆端口 107 发送的串行数据。

在数据接收中，端口逻辑块 104 和接收时钟(RXCLK 1)一起将经电缆端口 107 收到的数据包 DATA 1 经过选择块 103 发送给物理层逻辑块 102。如果端口逻辑 104 被选择块 103 选中，数据包(DATA 1)将写入物理层逻辑块 102 25 的 FIFO 内存中。

端口逻辑块 105 经过电缆端口 108 发送/接收仲裁信号(ARB.SIGNAL)和数据(DATA 2)，并具有从电缆端口 108 发送的数据信号和它的选通信号中产生接收时钟(RXCLK 2)的功能。在仲裁过程中端口逻辑块 105 由物理层逻辑块 102 馈给一仲裁信号(ARB.SIGNAL)。

30 在数据传输时间内，端口逻辑块 105 按时钟发生块 110 提供的传输时钟(TXCLK)将从物理层逻辑块 102 经选择块 103 发送的数据包(DATA 2)转换成



通过电缆端口 108 发送的串行数据。

在数据接收时间内，端口逻辑块 105 和接收时钟(RXCLK 2)一起将由物理层逻辑块 102 通过电缆端口 108 经选择块 103 发送的数据包(DATA 2)发送到物理层逻辑块 102。如果此物理层逻辑块 102 被选择块 103 选中，数据包
5 数据(DATA 2)将被接收时钟(RXCLK 2)写入物理层逻辑块 102 的 FIFO 内存中。

端口逻辑块 106 通过电缆端口 109 发送/接收仲裁信号(ARB.SIGNAL)和数据(DATA 3)，并具有从电缆端口 109 发送的数据信号和它的选通信号中产生接收时钟(RXCLK 3)的功能。在仲裁过程中此端口逻辑块 106 由物理层逻辑块 102 馈给一仲裁信号(ARB.SIGNAL)。
10

在数据传输时间内，端口逻辑块 106 按时钟发生块 110 提供的传输时钟(TXCLK)将从物理层逻辑块 102 经选择块 103 发送的数据包(DATA 3)转换成通过电缆端口 109 发送的串行数据。

在数据接收时间内，端口逻辑块 106 和接收时钟(RXCLK 2)一起将经电缆端口 109 和选择块 103 接收到的数据包(DATA 3)发送到物理层逻辑块 102。如果此端口逻辑块 104 被选择块 103 选中，数据包(DATA 1)将被接收时钟(RXCLK 1)写入物理层逻辑块 102 的 FIFO 内存中。
15

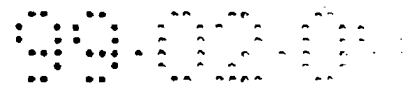
电缆端口 107 使用发自端口逻辑块 104 的信号驱动一双绞线电缆，同时转换通过双绞线电缆发送信号的级别以便将转换后的信号发送给端口逻辑块
20 104。

电缆端口 108 使用发自端口逻辑块 105 的信号驱动一双绞线电缆，同时转换通过双绞线电缆发送信号的级别以便将转换后的信号发送给端口逻辑块 105。

电缆端口 109 使用发自端口逻辑块 106 的信号驱动一双绞线电缆，同时
25 转换通过双绞线电缆发送信号的级别以便将转换后的信号发送给端口逻辑块 106。

时钟发生块 110 从由石英振荡器 111 提供的 24.576MHz 的时钟产生 49.152MHz 的系统时钟(SYSCLK)和 98.304MHz 的传输时钟(TXCLK)。

物理层中仲裁信号的逻辑值有三个：“1”，“0”和“Z”，并按照表 1
30 和 2 所示的规则产生，并按表 3 所示的规则解码。值“Z”表示驱动器的不工作状态。



两套双绞线 202 中的一套双绞线 TPA/TPA * 发送一选通信号 (STRB_TX), 并接收该选通信号(STRB_RX)。信号 STRB_TX , DATA_TX , STRB_ENABLE 和 DATA_ENABLE 用于产生仲裁信号 (ARB_A_RX, ARB_B_RX)。

5

表 1. 产生仲裁信号的规则

传输仲裁信号 A (ARB_A_TX)	驱动器		备注
	STRB_TX	STRB_ENABLE	
Z	-	0	TPA 驱动器：不工作
0	0	1	TPA 驱动器：工作；选通：低
1	1	1	TPA 驱动器：工作；选通：高

表 2：产生仲裁信号的规则

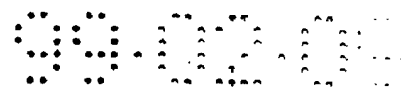
传输仲裁信号 B (ARB_B_TX)	驱动器		备注
	DATA_TX	DATA_ENABLE	
Z	-	0	TPB 驱动器：不工作
0	0	1	TPB 驱动器：工作；数据：低
1	1	1	TPB 驱动器：工作；数据：高

10

表 3：仲裁信号解码规则

接收仲裁比较值 (ARB_n_RX)	将从端口发送的仲裁信号(ARB_n_Rx)	插入仲裁信号(ARB_n)	备 注
			规定“n”为“A”或“B” 本表适用于两对信号
Z	Z	Z	如果此端口接收到 Z，收到的信号与电缆另一端端口发出的相同
0	Z	0	
1	Z	1	
Z	0	1	如果比较器在本端口发送 0 时收到 Z，则其它端口在发送 1
0	0	0	其它端口在发送 0 或 Z
Z	1	1	其它端口在发送 0
1	1	1	其它端口在发送 1 或 Z

而且在物理层中，两传输仲裁信号 (ARB_A_TX, ARB_B_TX) 按表 4 所示



的规则被编码成线性状态。如表 4 所示, 依据数据发送到父节点或子节点, 这些状态有不同的含义。

在 IEEE 1394 标准中解释了(节点的)父-子关系。在与网络连结的多个节点中, 有一些节点出现在末端(叶)。一旦重置后, 各节点立刻检查自己是否是叶。各节点是否是叶是以确认有多少电缆与该节点自身相连来检查的。具体地说, 只有一个端口的节点或有多个端口但仅有一个被连结的节点就是叶。各叶向与它相连的节点(父节点)进行一次查询。收到查询的节点接收与它相连的进行查询的节点作为子节点。尚未确定父-子关系的节点向连结终端作进一步查询。这样确定了网络中的父-子关系。最终, 在它们的全范围内作为父节点的节点成为根。

表 4 : 发自物理层的仲裁的线性状态

仲裁传输		线性状态名称	备注
ARB_A_TX	ARB_B_TX		
Z	Z	IDLE	发送以显示间隙
Z	0	TX_REQUEST 传输请求	发送给父节点以要求总线
		TX_GRANT 传输给予	当给予总线时发送给予节点
0	Z	TX_PARENT_NOTIFY 传输父节点通知	在 TREE_ID 状态发送给父节点竞争者
0	1	TX_DATA_PREFIX 传输数据前标	在数据包之前或在相连子操作的数据包之间发送
1	Z	TX_CHILD_NOTIFY 传输子节点通知	发送给子节点使了解 PARENT NOTIFY
		TX_IDENT_DONE 传输确认结束	发送给父节点说明 SELF_ID 状态结束
1	0	TX_DATA_END 数据传输结束	在数据包传输结束后发送
1	1	总线重置 BUS_RESET	为总线的重新设置而发送

在物理层中, 插入仲裁信号(ARB_A, ARB_B)按下面表 5 所示的规则被解码成线性状态。



表 5：物理层接收到的仲裁信号的线性状态

插入仲裁信号		线性状态名称	备 注
ARB_A	ARB_B		
Z	Z	IDLE 闲	连结的相邻节点的 PHY 未工作
Z	0	RX_PARENT_NOTIFY 接收父节点通知	连结的相邻节点的 PHY 试图成为子节点
Z	0	RX_REQUEST_CNCEL 接收请求取消	连结的相邻节点的 PHY 放弃了请求
Z	1	RX_IDENT_DONE 接收确认结束	子节点的 PHY 结束了 SELF_ID 状态
0	Z	RX_SELF_ID_GRANT 接收自确认给予	父节点的 PHY 为 SELF_ID 给予总线
		RX_REQUEST 接收请求	子节点的 PHY 请求总线
0	0	RX_ROOT_CONTENTION 接收根争夺	子节点的 PHY 和连接的相邻节点的 PHY 都试图成为子节点
		RX_GRANT 接收给予	父节点的 PHY 给予总线控制
0	1	RX_PARENT_HANDSHAKE 接收父节点握手	连结的相邻节点的 PHY 了解 PARENT_NOTIFY
0	1	RX_DATA_END 接收数据结束	连结的相邻节点的 PHY 结束数据块传输使总线变闲
1	Z	RX_CHILD_HANDSHAKE 接收子节点握手	连结的相邻节点的 PHY 了解 TX_CHILD_NOTIFY
1	0	RX_DATA_PREFIX 接收数据的前标	连结的相邻节点的 PHY 试图发送数据包或在数据块传输结束后试图发送数据
1	1	SBUS_RESET 总线重置	发送以便总线的重新设置

上面定义的 IEEE 1394 标准为一连接图像处理的内部设备到计算机的接口提供了必要的条件，这样很多设备，例如音频或视频设备或个人计算机，可以容易地用一根单独的电缆在一房间内构成家庭网以允许多种设备的简化

但是，由于 IEEE 1394 标准提供了互连设备之间的距离，即节点间电缆的最大距离为 4.5m，因此，如果要在多个房间内建设一个家庭网络，必须提供大量仅作为电缆中继的节点。

而且，如果希望延长电缆而不改变 IEEE 1394 标准的物理层系统，需要
5 增大电缆的直径，这就降低网络中电缆布线的可实施性，而且电缆本自很贵。

本发明的目的是提供一种进行数字串行数据传输的方法和装置，其中如 IEEE 1394 标准情形一样在数据传输前先进行总线使用权仲裁的数字串行数据接口的节点间的电缆距离被延长以便于长距离传输。

根据本发明，提供一种发送和接收数据和控制代码的装置，包括：一输入/输出端口；数据转换装置，用于将传输数据从 n-bit 代码转换成 m-bit 代码
10 和将接收的数据从 m-bit 代码转换成 n-bit 代码；控制信号转换装置，用于将获得与输入/输出端口相连的传输通道的使用权控制信号转换成由与分配给数据的 m-位码不同的 m-位码构成的控制代码并将接收到的 m-位代码转换成控制信号；加扰装置，用于对从控制信号转换装置接收到的 m-位控制代码进
15 行加扰以减少不必要的辐射，并将加扰后的控制信号输出给输入/输出端口；和解扰装置，用于对从输入/输出端口接收到的加扰后的信号进行解扰并将解扰的信号输出给控制信号转换装置。

根据本发明，还提供一种数据传输方法，包括：一发送数据转换步骤，用于把待传输的数据从 n-位码转换成 m 位码；一接收数据转换步骤用于把接
20 收的数据从 m 位码转换成 n 位码，一传输控制信号转换步骤，用于把获得与输入/输出端口相连的传输通道的使用权的控制信号转换成由与分配给数据的 m-位码不同的 m-位码构成的控制码及把接收的 m 位码转换成控制信号；一加扰步骤，用于对从传输控制信号转换步骤接收到的 m 位控制码进行加扰以减少不必要的辐射并将加扰后的控制信号输出给输入/输出端口；和一解扰
25 步骤，用于对从输入/输出端口接收到的加扰信号进行解扰。

图 1 是 IEEE 1394 标准中传输数据的信号结构的时间示意图。

图 2 是 IEEE 1394 标准规定的电缆截面视图。

图 3 给出一个应用 IEEE 1394 标准的网络的示意结构。

图 4 给出遵守 IEEE 1394 标准的一个协议结构和接口。

30 图 5 给出一个异步传输的包。

图 6 给出通过仲裁获得总线使用权状态。

图 7 给出一个等时传输的包。

图 8 为一框图，给出 IEEE 1394 标准中物理层的示意结构。

图 9 为一框图，给出用于本发明实施例中进行数字串行数据接口的接口器件的结构。

5 图 10 为一框图，给出用于上面数字串行数据的接口器件的扰码块的详细结构。

图 11 为一框图，给出上面接口器件中用于数字串行数据的解扰码块的详细结构的一部分。

图 12 为一框图，给出解扰码块详细结构的一部分。

10 图 13 为一框图，给出解扰码块详细结构的一部分。

图 14 为一框图，给出解扰码块详细结构的全部。

图 15 为一框图，给出解扰码块的另一个详细结构的全部。

参照示图，详述本发明的优选实施例。

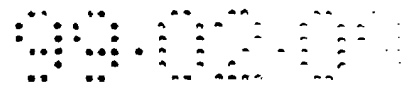
15 本发明通过使用具体形式如图 9 所示的用于数字串行数据的接口器件而实现的。

图 9 所示的接口器件，包括：一物理层逻辑块(PHY LOGIC)1；一个选择块(RXCLOCK/DATA SELECTOR)2；一个转换处理块(4B/5B CONVERTER & ARB.SIGNAL CONVERTER)3；加扰块(SCRAMBLE 1, SCRAMBLE 2)4A, 4B；解扰块(DE-SCRAMBLE 1, DE-SCRAMBLE 2)5A, 5B；传输块(P/S1, 20 P/S2)6A, 6B；接收块(RX_PLL 1P/S, RX_PLL 2P/S)7A, 7B；一端口逻辑块(PORT LOGIC)8，一模拟驱动器(ANALOG DRIVER)9；和一时钟发生块(PLL)10。

25 物理层逻辑块 1 利用 IEEE 1394 高效串行总线标准，以下称为 IEEE 1394 标准中提供的链接层进行 I/O 控制和仲裁控制，并与符合 IEEE 1394 标准的链接层控制器 100 和上述的选择块 2，转换处理块 3 及端口逻辑块 8 相连。

物理层逻辑块 1 中的链接层的 I/O 等同于 IEEE 1394 标准，即链接层和物理层之间的通信是通过发送/接收数据信号(DATA)和控制信号(CTRL)进行的。而且，链接层发送一连接请求信号(LREQ)给物理层逻辑块 1 作为从链接层到物理层的发送请求。

30 本物理层逻辑块 1 有一仲裁控制器，封装在其中，并由该仲裁控制器控制在仲裁过程和总线之间的传输/接收。如果从链接层产生一个包传输请求，



5 的节点中， $ABR_A = 0$ 且 $ABR_B = 0$ ，如表 3 所示。即，从相应节点发送的信号相互取消。在本实施例中，由于数字串行信号被送到双绞线，该取消没有进行。于是利用将要发送给物理层逻辑 1 的相关的仲裁信号，将传输码元存储在转换处理块 3 的存储器或寄存器中并且根据存储的传输码元和发
解扰块 5A, 5B 的接收码元设定仲裁状态。对于发送到物理层逻辑块 1 的仲裁信号，按照 IEEE 1394 标准分配 2 位信号以表示“0”，“1”和“Z”。相关的仲裁状态从传输和接收码元中建立而且根据表 8 产生仲裁信号并发送给物理层逻辑块 1。

一方面，也可以通过根据发送到一状态机器的码元管理节点状态来根据
10 节点的状态和收到的码元设定仲裁的状态，未示出。在本实施例中，表 7 相应于 IEEE 1394 标准中的表 5。

表 6 为仲裁分配的传输码元

传输码元	仲裁状态
11111	IDLE 空闲
00100	TX_REQUEST 传输请求
	TX_GRANT 传输给予
00101	TX_PARENT_NOTIFY 传输父节点通知
11000 10001	TX_DATA_PREFIX 传输数据前缀
00111	TX_CHILD_NOTIFY 传输子节点通知
	TX_IDENT_DONE 传输确认结束
01101	TX_DATA_END 传输数据结束
00000 11111	BUS_RESET 总线重置

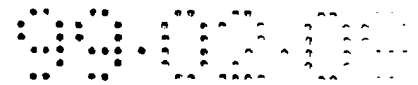
15 在数据包数据传输中，转换处理块 3 将经过选择块 2 发送的数据包数据 (DATA 1, DATA 2) 从 4 位信号转换成 5 位信号，分配如表 9 所示，以便把转换后的信号发送到加扰块 4A, 4B。同时，转换处理块 3 将发自解扰块 5A, 5B 的 5 位接收数据包数据从 5 位信号转换成 4 位信号，它们将被发送给选择块 2。



接收码元	传输码元	仲裁状态
11111	11111	IDLE 空闲
00101	11111	RX_PARENT_NOTIFY 接收_父节点_通知
11111	00100	RX_REQUEST_CANCEL 接收_请求_取消
00111	11111	RX_IDENT_DONE 接收_确认_结束
00100	11111	RX_SELF_ID_GRANT 接收_自_确认_给予
00100	11111	RX_REQUEST 接收_请求
00101	00101	RX_ROOT_CONTENTION 接收_根_争夺
00100	00100	RX_GRANT 接收_给予
00111	00101	RX_PARENT_HANDSHAKE 接收_父节点_握手
01101	11111	RX_DATA_END 接收_数据_结束
11111	00111	RX_CHILD_HANDSHAKE 接收_子节点_握手
11000 10001	00100	RX_DATA_PREFIX 接收_数据_前缀
11000 10001	00111	RX_DATA_PREFIX 接收_数据_前缀
11000 10001	11111	PX_DATA_PREFIX 接收_数据_前缀
00000 11111	(无关紧要)	BUS_RESET 总线_重置

表 8 物理层接收到的仲裁信号的线性状态

仲裁状态		仲裁信号	
		ARB_A	ARB_B
IDLE	空闲	Z	Z
RX_PARENT_NOTIFY	接收_父节点_通知	Z	0
RX_REQUEST_CANCEL	接收_请求_取消		
RX_IDENT_DONE	接收_确认_结束	Z	1
RX_SELF_ID_GRANT	接收_自_确认_给予	0	Z
RX_REQUEST	接收_请求		
RX_ROOT_CONTENTION	接收_根_争夺	0	0
RX_GRANT	接收_给予		



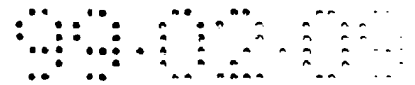
RX_PARENT_HANDSHAKE	接收_父节点_握手	0	1
RX_DATA_END	接收_数据_结束		
RX_CHILD_HANDSHAKE	接收_子节点_握手	1	Z
RX_DATA_PREFIX	接收_数据_前缀	1	0
BUS_RESET	总线_重置	1	1

表 9 分配给数据的码元

十六进制	二进制	码元
0	0000	11110
1	0001	01001
2	0010	10100
3	0011	10101
4	0100	01010
5	0101	01011
6	0110	01110
7	0111	01111
8	1000	10010
9	1001	10011
A	1010	10110
B	1011	10111
C	1100	11010
D	1101	11011
E	1110	11100
F	1111	11101

在上面的转换处理块 3 的 4 位/5 位转换中, 含有大量时钟元素的 5 位码元被分配给数据包数据(DATA 1, DATA 2), 如表 9 所示。这样, 数据包数据 (DATA 1, DATA 2) 的接收端能够通过 PLL 主动地从接收信号中产生接收时钟 (RXCLK 1, RXCLK 2)。

通过分配含有极多 IDLE(1111), 即时钟消息, 的 5 位码元给 IEEE 1394 标准仲裁的空闲状态, 即使在仲裁的空闲状态中接收方 PLL 的锁定状态也能



够保持，这样使仲裁能够被可靠地执行。

在本接口器件中，其中，在每个传输块 6A, 6B 中，如以后要解释的，5 位的发送信号从 5 位的传输信号被转换成串行数据并且 NRZ(不归零)数据被转换成 NRZI(不归零反转)数据，并且按三级别信号发送对数据造成的变化，具有最高频率的发送信号是 11111...和 IEEE 1394 标准的仲裁中分配给空闲状态的 5 位空闲码元(11111)相对应，这样该信号占用了传输时间的主要部分。如果传输速率为 100Mbps，当以 4B/5B 转换发送 125Mbps 信号时在 31.25MHz 频率处产生不需要的辐射。要消除该峰，只要预置的格式不连续重复就足够了。于是，提供了加扰块 4A, 4B 以消除那些不需要的辐射。

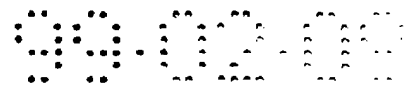
10 加扰块 4A, 4B 在数据包数据传输中对发自转换处理块 3 的 5 位传输信号加扰以消除该 5 位传输信号的不需要的辐射。传输块 6A, 6B 由加扰块 4A, 4B 馈给加扰的 5 位的传输信号。

而且，由于除空闲(11111)，TX_DATA_PREFIX(11000 10001)和 TX_DATA_END(01101)外码元都以 2 位“0”开始，按照这样的假定，如果在串行/并行转换后的信号同步时间内发现两位“0”，表示码元的起始端，15 则每个含有这些“0”的 5 比特的仲裁信号均可被确定为是码元。但是对于 BUS_RESET(00000 11111)，接收到连续的 5 个“0”不需要计数 2 位的“0”就可以确定每个仲裁信号。

为 BUS_RESET 分配两个码元(00000 11111)的原因是锁定 PLL 并保持该 20 锁定状态。如果只给 BUS_RESET 分配一个码元(00000)，则在接收 BUS_RESET 时无法锁住 PLL 并保持该锁定状态。在分配 5 位的码元给数据和仲裁状态时，选择这样的代码使在转换成 NRZI 码时 PLL 的锁定不在 DC 电平中产生偏压。由于满足这些条件的 5 位码元有确定的数目限止，可以通过组合两个码元以获得满足这些条件的代码。

25 对于 TX_DATA_PREFIX(11000 10001)，它的检测独立于其它仲裁信号。即把接收的数据移动一比特以准备五个数据串，每个数据串的长度为 10 比特。如果数据串与 TX_DATA_PREFIX(11000 10001)的比特型式吻合，确定接收 TX_DATA_PREFIX。由于数据包数据在接收 TX_DATA_PREFIX 之后紧接着收到，可以通过接收 TX_DATA_PREFIX 来对数据包数据施用代码同步。

30 由于 TX-DATA-END(01101)在数据包数据之后紧接着收到，它可以用与 TX-DATA-PREFIX 和数据包数据相同的码元同步来检测。而且，由于



TX_DATA_PREFIX 的比特型式(11000 10001)未出现在表 9 对数据包数据转换的数据串中, 即使不施用码元同步也无法在数据包数据中检测到, 所以不会出现接收的错误数据。在检测到 TX_DATA_PREFIX 后, 非 TX_DATA_END 或 BUS_RESET 的仲裁信号不被检测。

- 5 解扰块 5A, 5B 对发自接收块 7A, 7B 的 5 位信号进行与加扰块 4A, 4B 进行的加扰相连系的解扰以便对 5 位的接收信号解扰。转换处理块 3 由解扰块 5A, 5B 馈给解扰后的 5 位接收信号。

加扰块 4A, 4B 和解扰块 5A, 5B 的开/关状态的相应操作可以由开关设定。

- 10 传输块 6A, 6B 把由加扰块 4A, 4B 加扰后的加扰的 5 位传输信号从并行数据转换成串行数据并从 NRZ(不归零)数据转换成 NRZ1(不归零反转)以便按三值信号发送数据中引起的变化。

- 接收块 7A, 7B 也将接收信号从 NRZ1(不归零反转)转换成 NRZ(不归零)数据并把串行数据转换成并行数据以便将 5 位的接收数据发送给解扰块 5A, 15 5B。接收块 7A, 7B 通过 PLL 从接收到的数据中产生接收时钟(RXCLK 1, RXCLK 2)并将产生的时钟发送给选择块 2。

- 端口逻辑块 8 适用于发送/接收数据(DATA 3)和符合 IEEE 1394 标准的物理层的仲裁信号(ARB.SIGNAL 3)并且从发送自模拟驱动器 9 的数据和它的选通信号中产生接收时钟(RXCLK 3)。在仲裁中, 端口逻辑块 8 由物理层逻辑块 1 馈给仲裁信号(ARB.SIGNAL 3)。

在数据发送中, 端口逻辑块 8 通过发自时钟产生块 10 的发送时钟(TXCLK)将经过选择块 2 的发自物理层逻辑块 1 的数据包(DATA 3)转换成为将从模拟驱动器 9 发送的串行数据。

- 在数据接收中, 端口逻辑块 8 经过选择块 3 向物理层逻辑块 1 发送收自 25 模拟驱动器 9 的数据包数据(DATA 3)和接收时钟(RXCLK 3)。如果本端口逻辑块 8 被选择块 3 选中, 数据包(DATA 3)被接收时钟(RXCLK 3)写入物理层逻辑块 1 中的 FIFO 内存中。

时钟发生块 10 适用于从石英振荡器 11 给予的 24.576MHz 时钟中产生 49.152MHz 的系统时钟(SYSCLK)和 98.304MHz 的发送时钟(TXCLK)。

- 30 上述的串行数据接口器件包括转换处理块 3, 用于对仲裁信号(ARB.SIGNAL 1, ARB.SIGNAL 2)和数据包(DATA 1, DATA 2)进行 4 位/5 位转

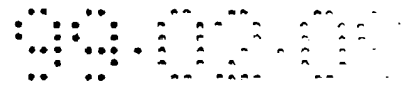
换以便通过传输块 6A, 6B 和接收块 7A, 7B 以 5 位码数据发送或接收仲裁信号(ARB.SIGNAL 1, ARB.SIGNAL 2)和数据包(DATA 1, DATA 2)来实现使用光纤电缆或不昂贵的 UTP(非屏蔽双绞线)电缆作为传输电缆的长距离传输。本接口器件还包括符合 IEEE 1394 标准的端口逻辑块 8 和模拟驱动器 9 以便按符合 IEEE 1394 关于光纤电缆的标准的电缆提出可能的共存的传输通道或者按 UTP 电缆(提出可能共存的)传输通道。

下面说明上面提到的加扰块 4A, 4B 和解扰块 5A, 5B。

上面的两个加扰块 4A, 4B 具有相同的结构而且每个包括密钥流寄存器 11A, 11B, 11C, 异或门 12A, 12B, 12C, 12D 和 12E, 异或门 13A, 13B, 13C, 13D, 13E 及一个传输时钟发生器 14, 在图 10 中给出例子。如果没有区分的必要时, 以后将密钥流寄存器 11A, 11B, 11C, 异或门 12A, 12B, 12C, 12D 和 12E 及异或门 13A, 13B, 13C, 13D 和 13E 分别称为密钥流寄存器 11, 异或电路 12 或异或电路 13。

在如图 10 给出的加扰块的示意结构中, 发送时钟(CLOCK)从发送时钟发生器 14 被发送给密钥流寄存 11。密钥流寄存器 11 是 11 位寄存器并且它的初始值被设为一非零的适当值, 例如[11111111111]。在密钥流寄存器 11 中, 密钥流寄存器 11A 的输出端口 OUT 0 到 OUT 4 的输出信号被发送到密钥流寄存器 11B 的输入端口 1N5 到 1N9, 每个发送时钟(CLOCK)使得密钥流寄存器 11B 的输出端口 OUT 5 的输出信号被发送到密钥流寄存器 11C 的输入端口 1N10 中。

另一方面, 密钥流寄存器 11A 的输出端口 OUT 4 的输出信号和密钥流寄存器 11B 的输出端口 OUT 6 的输出信号被发送到异或电路 12A, 其输出信号被发送到密钥流寄存器 11A 的输入端口 IN0 和异或电路 13A 中。密钥流寄存器 11B 的输出端口 OUT 5 和 OUT 7 的输出信号进入异或电路 12B, 其输出信号被发送到密钥流寄存器 11A 的输入端口 IN1 和异或电路 13B 中。密钥流寄存器 11B 的输出端口 OUT 6 和 OUT 8 的输出信号进入异或电路 12C, 其输出信号被发送到密钥流寄存器 11A 的输入端口 IN2 和异或电路 13C 中。另一方面, 密钥流寄存器 11B 的输出端口 OUT 7 和 OUT 9 的输出信号进入异或电路 12D, 其输出信号被发送到密钥流寄存器 11A 的输入端口 1N3 和异或电路 13D 中。密钥流寄存器 11B 的输出端口 OUT 8 的输出信号和密钥流寄存器 11C 的输出端口 OUT 10 的输出信号被发送到异或电路 12E 中, 其



输出信号被发送到密钥流寄存器 11A 的输入端口 IN4 和异或电路 13E 中。

如果密钥流寄存器 11 内容的初值为“ABCDEFGHIJK”，其中 A 到 K 代表 1 或 0，A 是密钥流寄存器 11C 的输出，BCDEF 是密钥流寄存器 11B 的输出而 GHIJK 是密钥流寄存器 11A 的输出，密钥流寄存器 11 的内容在每个传输时钟(CLOCK)都被移动，如表 10 所示，其中 \cap 表示异或输出。

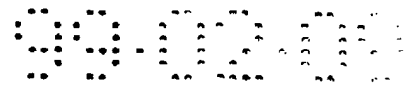
在本加扰块中，5 位的输入信号 PLAIN DATA(BIT 0, BIT 1, BIT 2, BIT 3, BIT 4) 作为输入信号被发送到异或电路 13 中以便通过此异或电路 13 输出 5 位的输出信号 SCRAMBLED DATA。即 5 位的输出信号 SCRAMBLED DATA 是从 5 位的输入信号 PLAIN DATA(BIT 0, BIT 1, BIT 2, BIT 3, BIT 4) 的异或输出获得的并且异或电路 12 的输出信号被发送到输入端口 IN0 到 IN4。

两个加扰块 5A, 5B 具有相同的结构，如图 11 到 14 所示，且包括：加扰的数据寄存器 21A, 21B, 21C；虚拟(hypothesis)寄存器 22A, 22B, 22C；一比较格式寄存器 23；一虚拟数据比较器 24；密钥流寄存器 25A, 25B, 25C；一检测格式寄存器 26；一 2 位计数器 27；密钥流选择器 28A, 28B, 28C；一接收时钟发生器 29；异或电路 30A, 30B, 30D, 30E；异或电路 30A, 30B, 30C, 30D, 30E；异或电路 31A, 31B, 31C, 31D, 31E；异或电路 32A, 32B, 32C, 32D, 32E；异或电路 33A, 33B, 33C, 33D, 33E 及一异或电路 34。如果不需要区分时，加扰的数据寄存器 21A, 21B, 21C，虚拟寄存器 22A, 22B, 22C，密钥流寄存器 25A, 25B, 25C，密钥流选择器 28A, 28B, 28C，异或电路 30A, 30B, 30C, 30D, 30E，异或电路 31A, 31B, 31C, 31D, 31E，异或电路 32A, 32B, 32C, 32D, 32E，和异或电路 33A, 33B, 33C, 33D, 33E 以后分别被称为加扰的数据寄存器 21，虚拟寄存器 22，密钥流寄存器 25，密钥流选择器 28，异或电路 30，异或电路 31，异或电路 32，和异或电路 33。

尽管为了示意而分开，图 11 到 13 实际上是相互连接的。图 11 中的 A, B, C, D, E, 和 F 分别与图 12 中的 A', B', C', D', E', 和 F' 相连。类似的，图 11 中的 G, H, 和 I 分别与图 13 中的 G', H' 和 I' 相连，而图 13 中的 J 和 K 与图 11 中的 J' 和 K' 分别相连。图 14 是其整体结构简图。

Table 10

[illegible]



在此解扰块的本示意例子中，收到的 5 位加扰的数据被发送到如图 11 所示的异或电路 31，被发送到如图 12 所示的异或电路 33，还被发送到加扰的数据寄存器 21A 的输入端口 IN0 到 IN4。

由发送时钟发生器 29 向加扰的数据寄存器 21，虚拟寄存器 22 并向 2 5 位计数器 27 发送接收时钟(CLOCK)，如图 11 所示。接收到的 5 位加扰的数据被存入加扰的数据寄存器 21A。

按照来自加扰数据寄存器的每个接收时钟(CLOCK)，加扰数据寄存器 21 中的加扰数据寄存器 21A 的输出端口 OUT 0 到 OUT 4 的输出信号被发送到加扰的数据寄存器 21B 的输入端口 IN5 到 IN9，而加扰的数据寄存器 21B 的 10 输出端口 OUT 5 的输出信号被发送到加扰的数据寄存器 21 中的加扰数据寄存器 21C 的输入端口 IN0 中。

加扰数据寄存器 21A 的输出端口 OUT 4 的输出信号和加扰数据寄存器 21B 的输出端口 OUT 6 的输出信号被发送到异或电路 30A 中，其输出信号被发送到异或电路 31A。加扰数据寄存器 21B 的输出端口 OUT 5 和 OUT 7 的 15 输出信号被发送到异或电路 30B 中，其输出信号被发送到异或电路 31B 中。同样，加扰数据寄存器 21B 的输出端口 OUT 6 和 OUT 8 的输出信号被发送到异或电路 30C 中，其输出信号被发送到异或电路 31C 中。相似地，加扰数据寄存器 21B 的输出端口 OUT 7 和 OUT 9 的输出信号发送到异或电路 30D 中，其输出信号被发送到异或电路 31D 中。此外，加扰数据寄存器 21B 的 20 输出端口 OUT 8 的输出信号和加扰数据寄存器 21C 的输出端口 OUT 10 的输出信号被发送到异或电路 30E 中，其输出信号进入异或电路 31E。加扰数据寄存器 21 的输出端口 OUT 0 到 OUT 10 的输出信号被发送到图 13 所示的一个 11 位异或电路 34 中。

在解扰块中，收到的 5 位加扰数据作为输入信号被发送到加扰数据寄存器 21A 的输入端口 IN0 到 IN4 和异或电路 31 中，将要通过异或电路 31 被发 25 送到虚拟寄存器 22A 的输入端口 IN0 到 IN4 中。即一个与接收到的 5 位加扰数据和由加扰数据寄存器 21A 的输出端口 OUT 4 到 OUT 10 的输出信号馈给的异或电路 30 的输出信号的逻辑和对应的 5 位虚拟数据(HYPOTHESIS DATA)被发送到虚拟寄存器 22A 的输入端口 IN0 到 IN4 中。

30 虚拟寄存器 22 是一个 11 位的寄存器，其如此设计使得每从接收时钟发生器 29 发送接收时钟(CLOCK)时，虚拟寄存器 22A 的输出端口 OUT 0 到 OUT



4 的输出信号被发送到虚拟寄存器 22B 的输入端口 IN5 到 IN9 中，并使得虚拟寄存器 22B 的输出端口 OUT 5 的输出信号被发送到虚拟寄存器 22C 的输入端口 IN10 中。如图 13 所示，虚拟寄存器 22 的输出端口 OUT 0 到 OUT 10 的输出信号被发送到虚拟数据比较器 24 中。

- 5 如果 5 位虚拟数据(HYPOTHESIS DATA)在时刻 t 为 H0(t), H1(t), H2(t), H3(t)和 H4(t), 5 位明码(plain)数据 PLAIN DATA 为 P0(t), P1(t), P2(t), P3(t), 和 P4(t), 且密钥流寄存器 11 的值为[abcdefghijk], 下列关系:

$$\begin{aligned} H0(t+3) &= I \sim B \sim P0(t+3) \sim K \sim B \sim D \sim P4(t+2) \sim I \sim K \sim P1(t+1) \\ &= P0(t+3) \sim P4(t+2) \sim P1(t+1) \end{aligned}$$

10
$$\begin{aligned} H1(t+3) &= H \sim A \sim C \sim P1(t+3) \sim J \sim A \sim C \sim P0(t+1) \sim H \sim J \sim P2(t+1) \\ &= P1(t+3) \sim P0(t+1) \sim P2(t+1) \end{aligned}$$

$$\begin{aligned} H2(t+3) &= G \sim K \sim P2(t+3) \sim I \sim K \sim P2(t+1) \sim G \sim I \sim P3(t+1) \\ &= P2(t+3) \sim P1(t+1) \sim P3(t+1) \end{aligned}$$

15
$$\begin{aligned} H3(t+3) &= F \sim J \sim P3(t+3) \sim H \sim J \sim P2(t+1) \sim F \sim H \sim P4(t+1) \\ &= P3(t+3) \sim P2(t+1) \sim P4(t+1) \end{aligned}$$

$$\begin{aligned} H4(t+3) &= E \sim I \sim P4(t+3) \sim G \sim I \sim P3(t+1) \sim E \sim G \sim P0(t) \\ &= P4(t+3) \sim P3(t+1) \sim P0(t) \end{aligned}$$

20

在虚拟数据(HYPOTHESIS DATA)和明码数据(PLAIN DATA)之间成立，其中 \sim 表示异或。

- 即利用虚拟数据(HYPOTHESIS DATA)，密钥流元素将在所有时间内相互消除，不论发送端的密钥流寄存器 11 的初始值和时间，作为上面运算的结果，所以输出是明码数据(PLAIN DATA)的运算结果。

利用上面的关系，可以在虚拟数据比较器 24 中检测 IDLE(11111)和 BUS RESET(00000 11111)。

注明了当虚拟寄存器 22 的输出值为 “ 1111111111 ” 时可以检测 IDLE(11111)。

- 30 要检测 BUS_RESET，必须检测各格式，即(00000 11110), (00001 11110), (00011 11100), (00111 11000), (01111 10000), (11111 00000), (11110 00001),



(11100 00011), (11000 00111)和(10000 01111)。此时虚拟寄存器 22 的输出值为
[01000101110], [10001011101], [00010111010], [00101110100], [01011101000],
[10111010001], [01110100010], [11101000101], [11010001011] 和
[10100010111]。比较格式寄存器 23 用于保持用于虚拟数据比较器 24b 以检
5 测用于检测 BUS_RESET(00000 11111)的 IDLE(11111)和比较格式
[01000101110], [10001011101], [00010111010], [00101110100], [01011101000],
[10111010001], [01110100010], [11101000101], [11010001011] 和
[10100010111], 的比较格式[11111111111]。

虚拟数据比较器 24 对虚拟寄存器 22 的输出数据和比较格式寄存器 23
10 给予的 11 种比较格式进行比较。如果与任何比较格式重合, 虚拟数据比较器
24 输出一向上计数信号给 2 位计数器 27, 同时输出说明与哪个比较格式重
合的信息给检测格式寄存器 26。当装入信号已经进入, 或者假如虚拟寄存器
22 的输出数据未与比输格式寄存器 23 给予的 11 种比较格式重合, 虚拟数据
比较器 24 发出一个重置信号给 2 位计数器 27。每次当向上计数信号进入时
15 2 位计数器 27(加 1)。如果计数值为[3], 2 位计数器 27 输出一装入信号给虚
拟数据比较器 24 和密钥流选择器 28。如果被馈给一重置信号, 2 位计数器
27 将计数器重置为[0]。这样只有当 IDLE 或 BUS_RESET 被连续检测到三次
同步才被施用。结果是减少了例如由于噪声造成的错误的位反转而产生的错
误同步化的可能性。

20 根据来自虚拟数据比较器 24 的说明哪个比较格式显示出重合的消息,
检测格式寄存器 26 在检测 IDLE 时, 输出检测格式[11111111111], 如表 11
所示; 在检测 BUS_RESET 时, 输出检测格式[10000011111], [00000111111],
[00001111110], [00011111100], [0011111100], [0111110000], [1111100000],
[1111000001], [1110000011]和[1100000111]给异或电路 34, 如表 12 所示。

25

表 11

比较格式值	检测格式值
[11111111111]	[11111111111]

表 12

比较格式值	检测格式值
-------	-------

[01000101110]	[1000001111]
[10001011101]	[0000011111]
[00010111010]	[0000111110]
[00101110100]	[0001111100]
[01011101000]	[0011111000]
[10111010001]	[0111110000]
[01110100010]	[1111100000]
[11101000101]	[1111000001]
[11010001011]	[1110000011]
[10100010111]	[1100000111]

异或电路 34 对加扰数据寄存器 21 的输出数据和检测格式寄存器 26 提供的检测格式取异或并将异或 OR 输出作为一个新的 11 位密钥流发送到密钥流选择器 28 的输入端口 B0 到 B10，如图 12 所示。

如果装入信号被从 2 位计数器 27 发送到那里，密钥流选择器 28A 的输入端口 A0 到 A4 由异或电路 32 的输出信号馈给并输出输入端口 130 到 134 的输入信号，即一个新的 11 位密钥流的 5 位在输出端口 S0 到 S4，如图 13 所示。如果没有馈给装入信号，密钥流选择器 28A 在输出端口 S0 到 S4 输出来自输入端口 A0 和 A1 的输入信号。密钥流选择器 28A 的输出端口 S0 到 S4 的输出信号进入密钥流寄存器 25A 的输入端口 IN0 到 IN4。密钥流寄存器 25A 如此设计使得每次接收时钟发生器 29 给予接收时钟时将输出端口 OUT 0 到 OUT 4 的输出信号发送到密钥流选择器 28B 的输入端口 A5 到 A9。密钥流寄存器 25A 的输出端口 OUT 4 的输出信号和密钥流寄存器 25B 的输出端口 OUT 6 的输出信号进入异或电路 32A，其输出信号被发送到密钥流选择器 28A 的输入端口 A0 和异或电路 33A。

如果从 2 位计数器 27 馈给装入信号，密钥流选择器 28B 输出输入端口 B5 到 B9 的输入信号，即一个新的 11 位密钥流在输出端口 S5 到 S9 的 5 位。如果没有馈给装入信号，密钥流选择器 28B 在输出端口 S5 到 S9 输出来自输入端口 A5 到 A9 的输入信号。密钥流选择块 28B 的输出端口 S5 到 S9 的输出信号被发送到密钥流寄存器 25B 的输入端口 IN5 到 IN9，其每接收时钟 (CLOCK) 时在输出端口 OUT 5 到 OUT 9 输出输出信号。密钥流寄存器 25B 的输出端口 OUT 5 的输出信号被发送到密钥流选择器 28C 的输出端口

IN10。密钥流寄存器 25B 的输出端口 OUT 5 和 OUT 7 的输出信号被发送到异或电路 32B，其输出信号被发送到密钥流选择器 28A 的输入端口 IN1 及异或电路 33B。密钥流寄存器 25B 的输出端口 OUT 6 和 OUT 8 的输出信号进入异或电路 32C，其输出信号进入密钥流选择器 28A 的输入端口 IN2 及异或电路 33D。密钥流寄存器 25B 的输出端口 OUT 5 和 OUT 7 的输出信号被发送到异或电路 32D，其输出信号被发送到密钥流选择器 28A 的输入端口 IN3 及异或电路 33D。密钥流寄存器 25B 的输出端口 OUT 8 的输出信号和密钥流寄存器 25C 的输出端口 OUT 10 的输出信号进入异或电路 32E，其输出信号进入密钥流选择器 28A 的输入端口 IN4 及异或电路 33E。

当从 2 位计数器 27 馈给装入信号时，密钥流选择器 28C 输入输出端口 B10 的输出信号，即为新的 11 位密钥流的一位到输出端口 S10，当没有馈给装入信号时，密钥流选择器 28C 输出来自输入端口 A10 的输入信号给输出端口 S10。密钥流选择器 28B 的输出端口 S10 的输出信号进入密钥流寄存器 25C 的输入端口 IN10，其在每次接收时钟(CLOCK)时发出输出端口 OUT 10 的输出信号给异或电路 32E。

在上述的解扰块中，装入信号从 2 位计数器 27 进入到密钥流选择器 28，一个来自异或电路 34 的新的 11 位密钥流通过密钥流选择器 28 被装入密钥流寄存器 25 以获得与传输端加扰块的同步以便启动正常的解扰操作。等同于发送端加扰块的输入信号 PLAIN DATA 的输出信号 DESCRAMBLED DATA 可以通过异或电路 33 的异或输出获得。

这样，就可能利用图 9 所示的接口器件通过加扰块 4A、4B，和解扰块 5A、5B 防止不需要的辐射以实现利用 UTP 电缆长距离传输串行数据。

在如图 14 所示的解扰块中标明，如果虚拟寄存器 22 的输出信号为 [1111111111]，出现检测到 IDLE(11111)。另一方面，在检测 BUS_RESET 时，十种格式(0000 11111), (00001 11110), (00011 11100), (00111 11000), (01111 10000), (11111 00000), (11110 00001), (11100 00011), (11000 00111)和(10000 01111)，在该时刻，虚拟寄存器 22 的输出值为[01000101110], [10001011101], [00010111010], [00101110100], [01011101000], [10111010001], [01110100010], [11101000101], [11010001011], [10100010111]。但是如果数据具有明确的格式，相关的虚拟数据(HYPOTHESIS DATA)将不考虑加扰元素被唯一地确定。相反，对于虚拟数据(HYPOTHESIS DATA)明确的情形，虚拟数据

(HYPOTHESIS DATA)并未唯一地确定。

按图 15 所示设计解扰块 5A, 5B, 可以高精度地完成解扰。在图 15 所示的解扰块中, 与图 14 所示相同的组成部分被表示为相同的参考数码且没有具体地解释。

5 即当前的解扰块对以异或 OR 电路 34 对如图 14 所示的解扰块中加扰数据寄存 21 的输出数据和检测格式寄存器 26 提供的检测格式取异或的异或 OR 输出获得的 11 位密钥流进行第二次格式检测以提高密钥流检测精确性, 并且进一步还包括一异或电路 40, 一测试 (tentative) 密钥流寄存器 41, 一测试解扰寄存器 42, 一比较测试解扰格式寄存器 43 和一解扰格式比较器 44。

10 在本解扰块中, 以异或电路 34 的异或输出获得的 11 位密钥流被装入测试密钥流寄存器 41, 与 2 位计数器 27 的输出同步。异或电路 40 对接收到的 5 位加扰数据和测试密钥流寄存器 41 的内容取异或以便将异或运算的结果发送到测试解扰寄存器 42。

15 即本解扰块把从异或电路 34 的异或输出获得的 11 位密钥流发送到测试密钥流寄存器 41 以便与异或电路 40 的测试密钥流寄存器 41 中的内容取异或来对接收到 5 位加扰数据解扰。

如果虚拟寄存器 22 的输出值为 [1111111111], 数据很可能是 IDLE(11111)。而且如果虚拟寄存器 22 的输出结果为 [01000101110], [10001011101], [00010111010], [00101110100], [01011101000], [10111010001],
20 [01110100010], [11101000101], [11010001011] 或 [10100010111], 数据很可能是 BUS_RESET。

在解扰格式比较器 44 中测试解扰寄存器 42 的内容与比较测试解扰格式寄存器 43 给予的比测试解扰格式进行比较。

如果虚拟寄存器 22 的输出值为 [10111010001], 数据很可能是
25 BUS_RESET(11111 00000)。由解扰格式比较器 44 对相应于解扰后的接收到的 5 位加扰数据的测试解扰寄存器 42 的内容是否与 BUS_RESET (1111100000) 重合进行比较。

比较测试格式寄存器 43 在每次接收时钟(CLOCK)产生周期码作为比较测试格式, 与加扰块中的移位寄存器的方式相同。

30 如果测试解扰寄存器 42 的内容与比较测试解扰格式重合, 接收到的 5 位加扰数据被测试密钥流寄存器 41 中的内容, 即测试流正确地解扰, 数据被

测试密钥流这样与发送端的加扰块同步就可以获得正确的解扰并且可以利用在密钥流寄存器 25 上装入测试解扰寄存器 42 的内容作为新密钥流产生等同于发送端的加扰块的输入信号 PLAIN DATA 的输出信号 DESCRAMBLED DATA 作为异或电路 33 的异或输出。

- 5 如果测试解扰寄存器 42 的内容未与比较测试解扰格式重合, 即使虚拟寄存器 22 的输出值是[1011101000], 对接收到的 5 位加扰数据解扰获得的测试解扰寄存器 42 的内容也不是 BUS_RESET(11111 00000)。因此, 没有新的密钥流被装入密钥流寄存器 25。

10 这样, 利用本解扰块, 使用从如图 14 所示的解扰块的加扰数据寄存器 21 的输出数据和发自检测格式寄存器 26 的检测格式取异或的异或电路 34 的异或输出获得密钥流, 检验接收到的 5 位加扰数据是否被正确地解扰, 从而提高密钥流检测的准确性并实现解扰的准确性。

15 这样, 利用本实施例, 提供了一作为用于 4B/5B 数据转换的 4 位/5 位转换装置和用于给仲裁信号分配不同于分配给 4 位/5 位转换装置中数据的 5 位码元的 5 位码元的仲裁信号转换装置的转换处理块 3(4B/5B CONVERTER & ARB.SIGNAL CONVERTER), 仲裁信号和数据包可以按 5 位数据码发送和接收。而且, 利用本实施例, 提供了上述的 4 位/5 位转换装置, 加扰块 4A, 4B(SCRAMBLE 1, SCRAMBLE 2)和解扰块 5A, 5B(DESCRAMBLE 1, DESCRAMBLE 2)可以防止传输通道中不必要的辐射。因此, 利用本发明, 20 可以使用光纤电缆或不昂贵的 UTP 电缆作为传输电缆实现长距离传输。

而且, 如果 5 位码元[11111]包含 IDLE(11111), 即为 IEEE 1394 标准中最大量的仲裁的空闲状态的时钟信息, 即使在仲裁的空闲状态在接收端也可以保持 PLL 的锁定状态以实现仲裁的可靠性。

25 而且, 由于加扰/解扰装置可以实现 5 位并行处理, 处理可以使用比处理串行数据情形的操作速度低的逻辑(块)来完成。而且, 如果一个分配给总线重置状态的 5 位码元或者一个 IEEE 1394 高效串行总线标准中分配给总线重置状态的 5 位码元被加扰一解扰处理装置检测到以施用解扰同步, 有可能在开始加扰状态中实现连结, 断开和电源的开/关。由于总线在子操作间隙, 在 IEEE 1394 高效串行总线标准定义的称为“异步”的异步传输模式的 ack 间隙, 以及在保证传输区的称为“等时”的实时传输模式的 iso 间隙中都处于 30 空闲状态, 可以通过由加扰/解扰处理装置检测空闲状态在间隙时间点施加再

同步以施用于解扰同步。

- 而且利用加扰/解扰处理装置, 由于仅在检测到分配给总线重置状态的 5 位码元或者分配给空闲状态的 5 位码元时施加再同步, 逻辑结构在尺寸上可以减小。而且加扰/解扰处理装置在检测该 5 位码地数次以后才使解扰同步,
- 5 消除了因为噪声造成的错误的位反转产生的错误同步以保证可靠的处理。

说明书附图

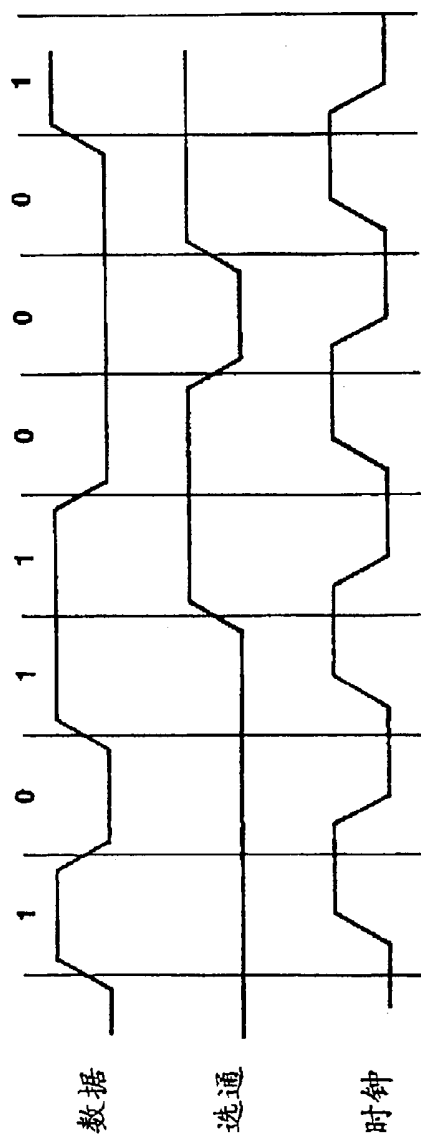


图 1

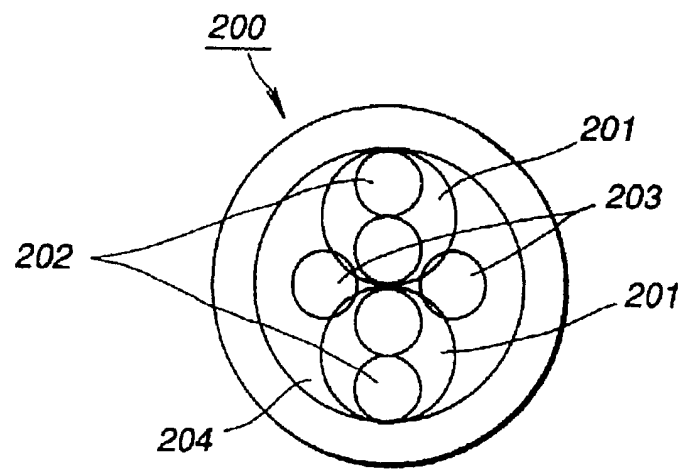


图 2

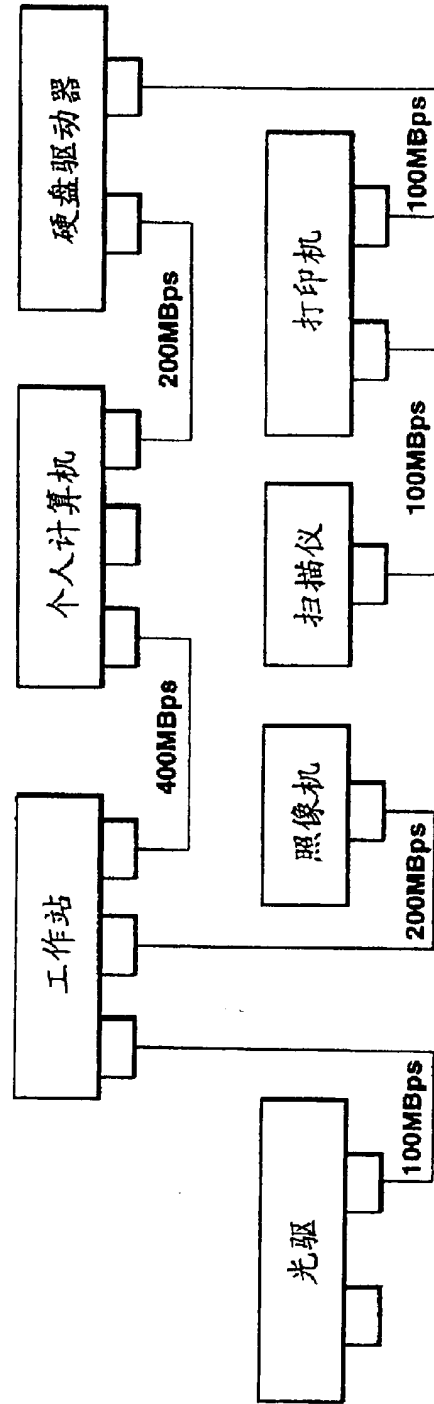


图 3

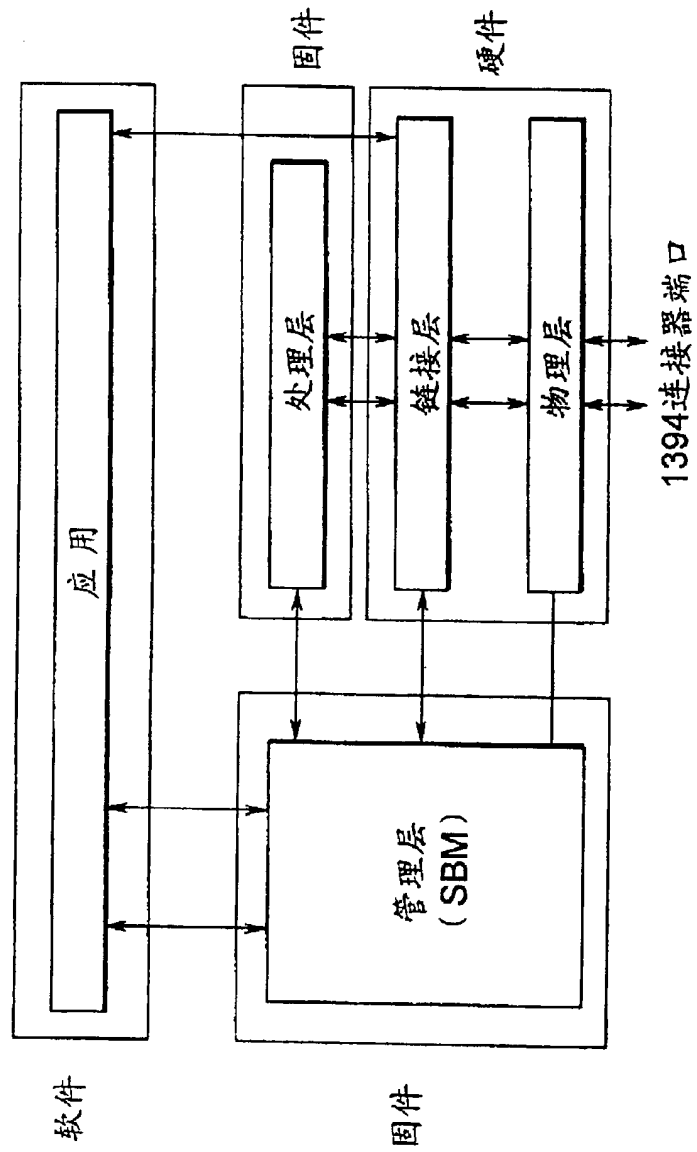
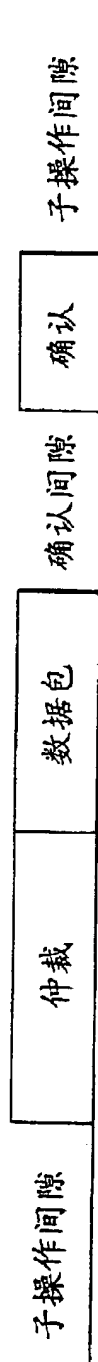


图 4



异步传输数据包

图 5

99.00.00

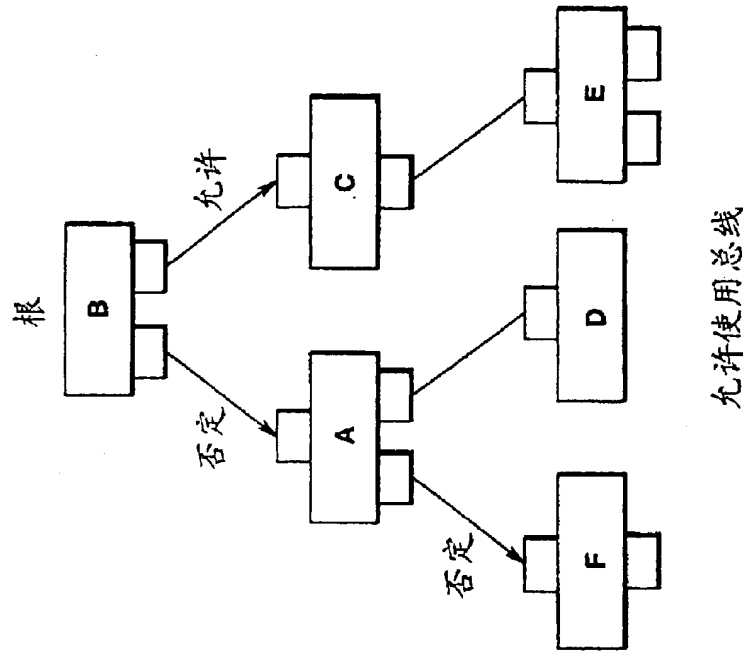


图 6b

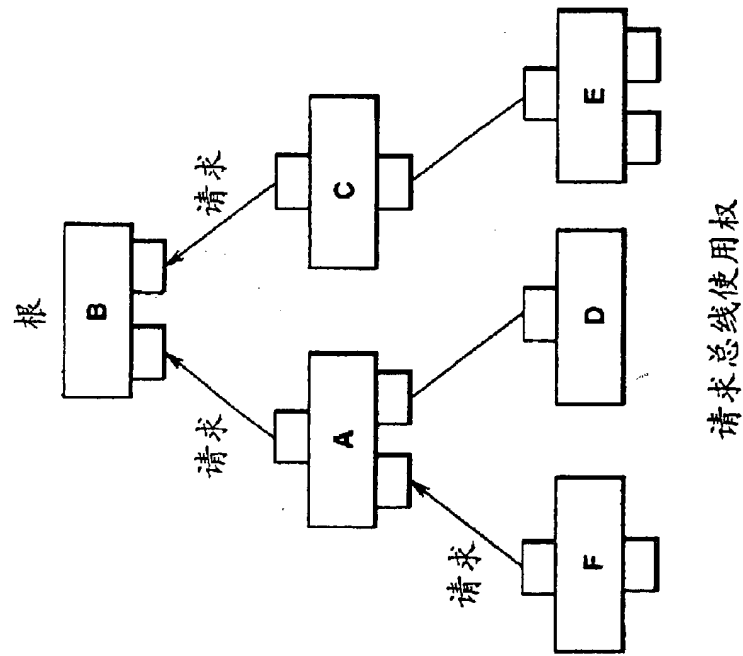
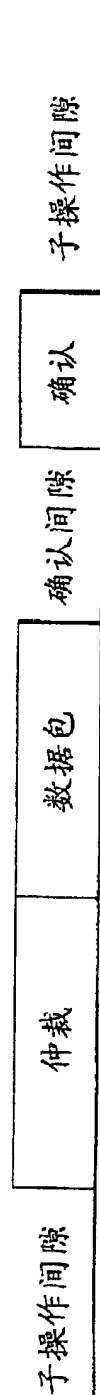


图 6a



通道 A 通道 B 通道 C

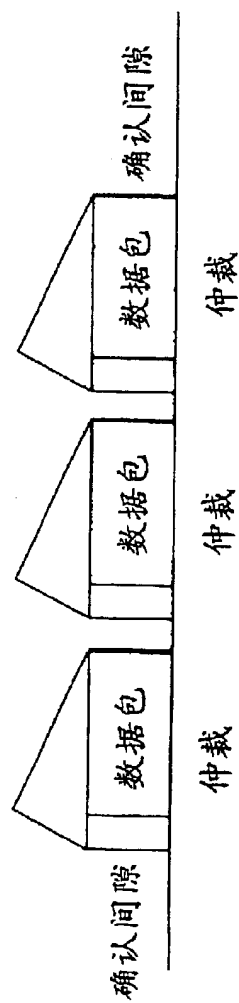


图 7

99.00.00

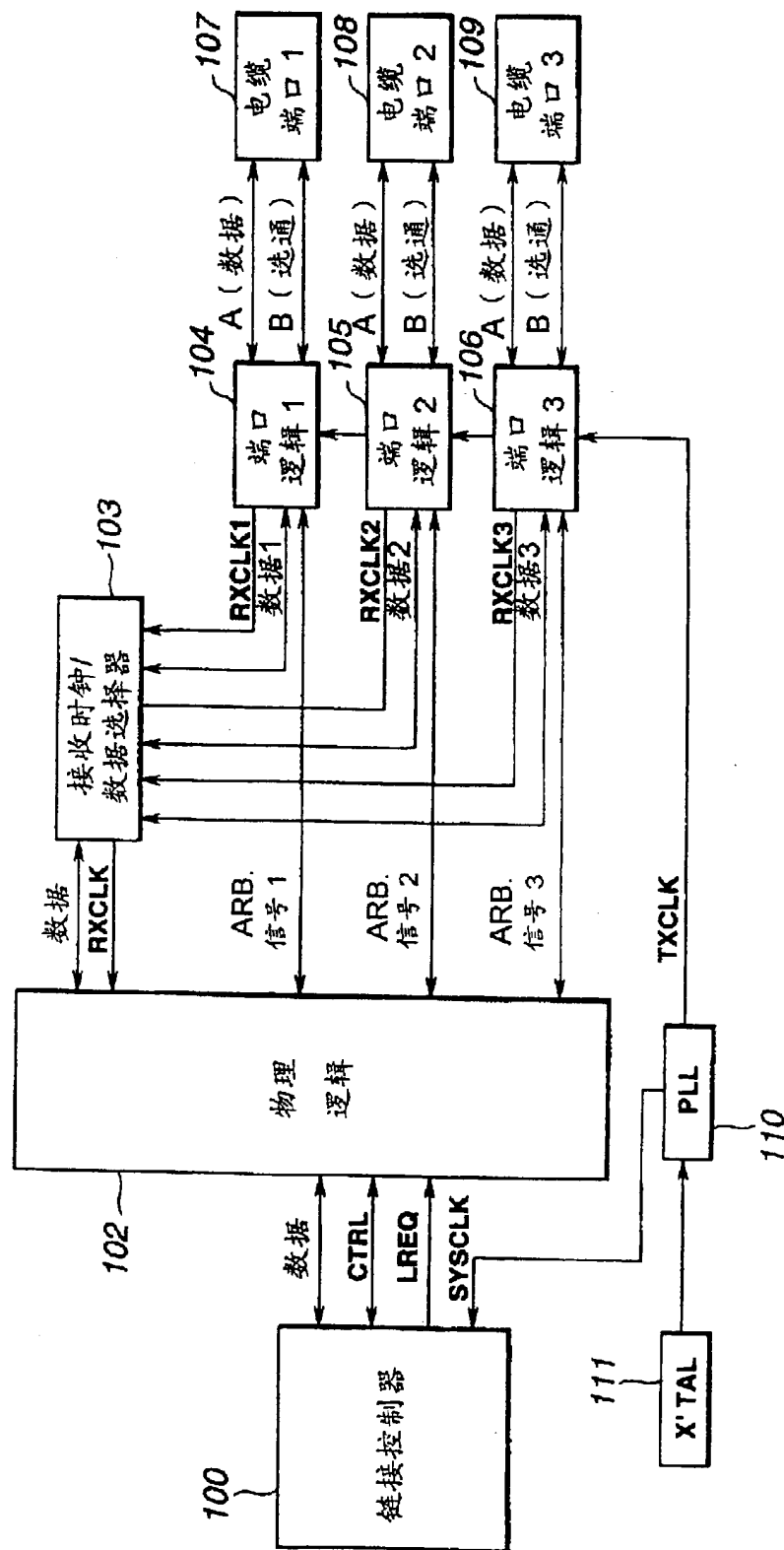


图 8

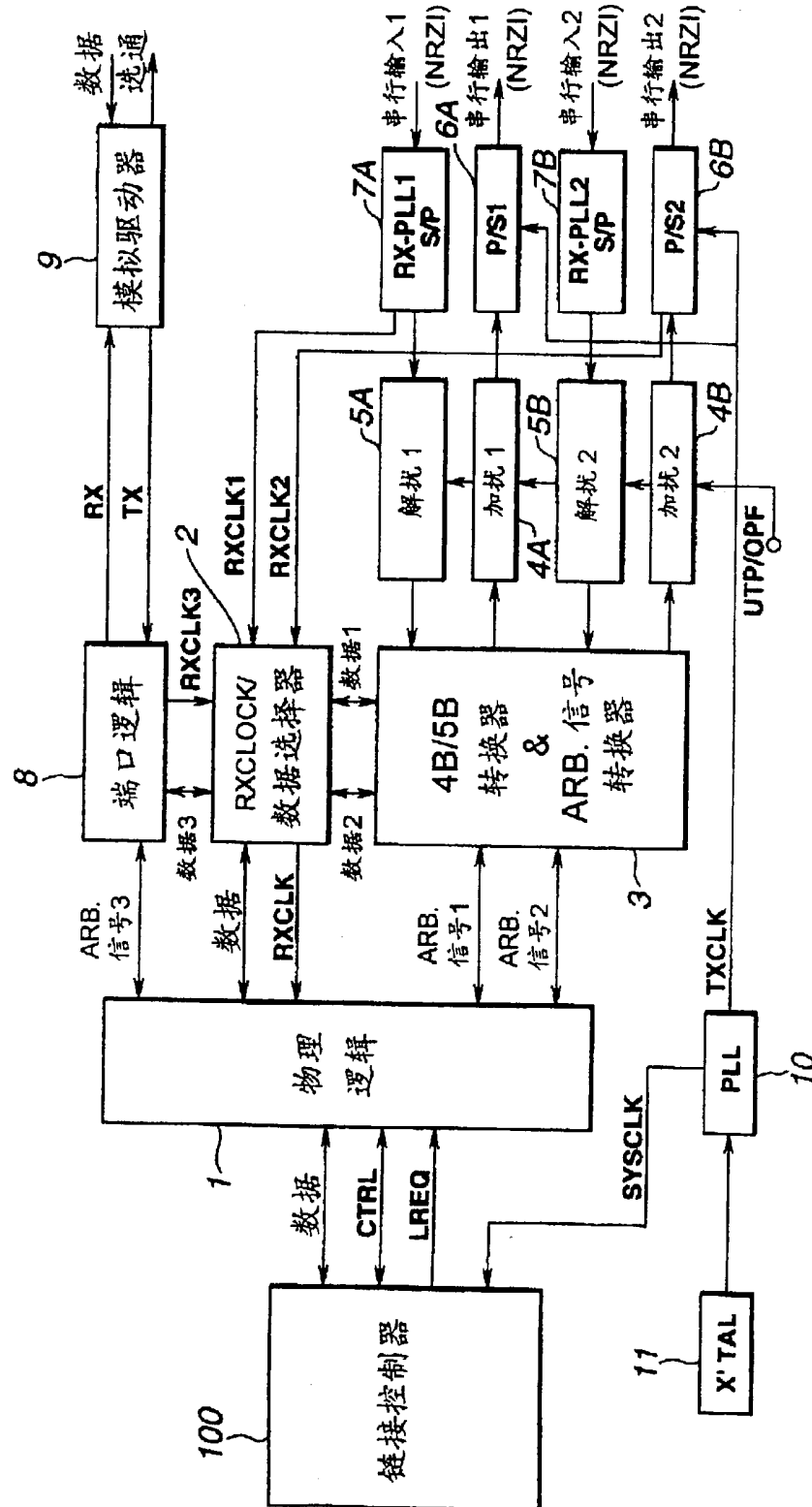


图 9

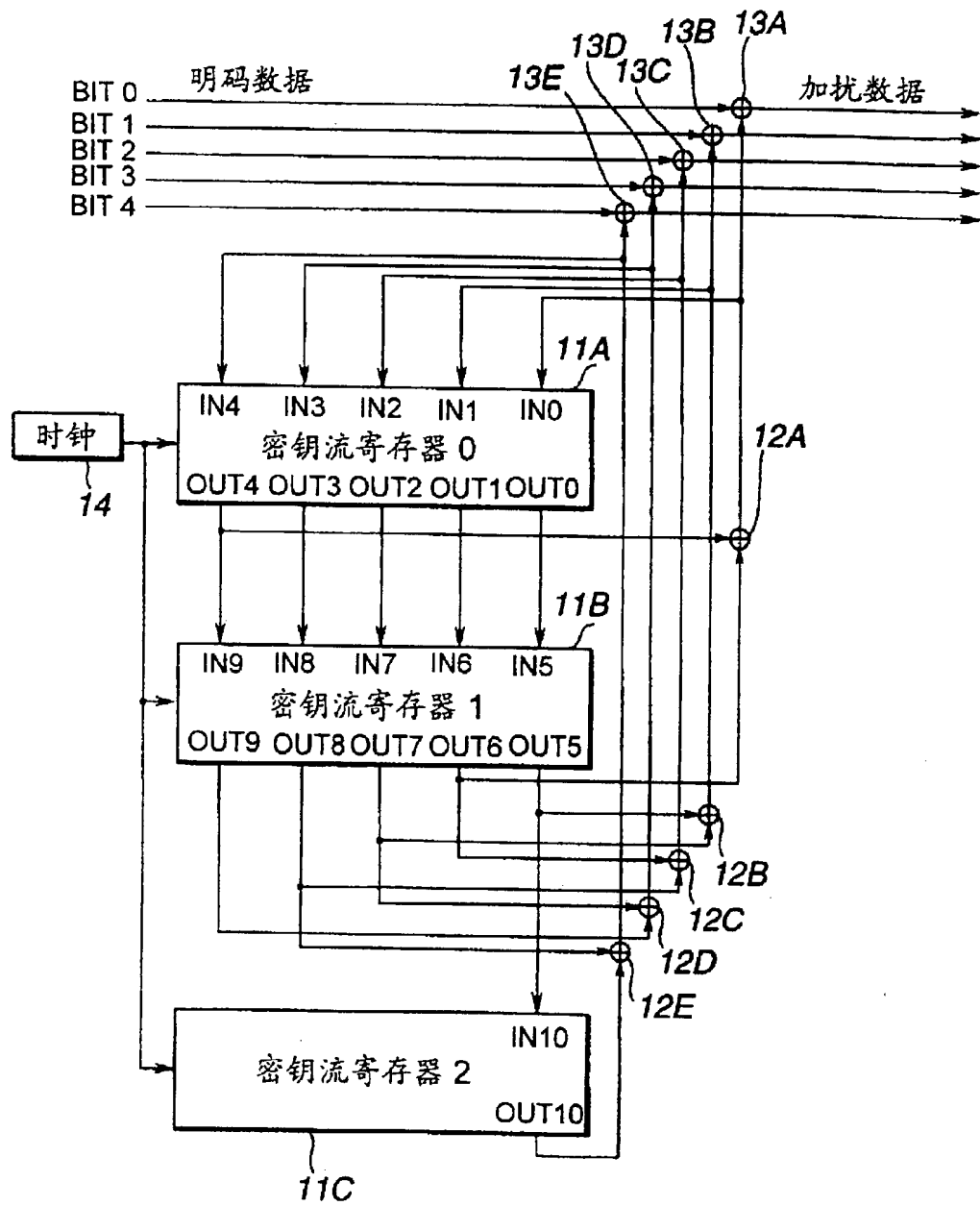


图 10

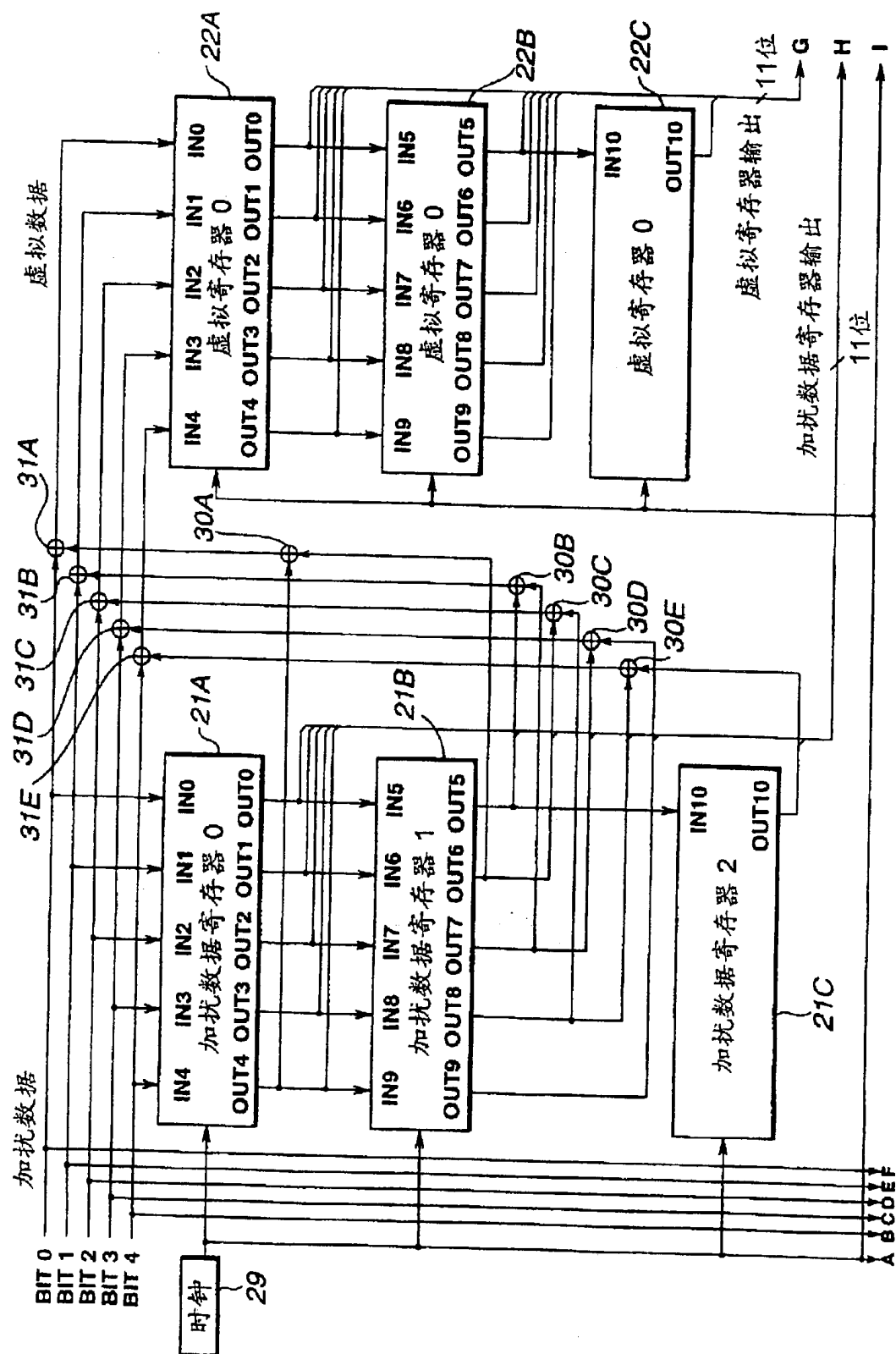


图 11

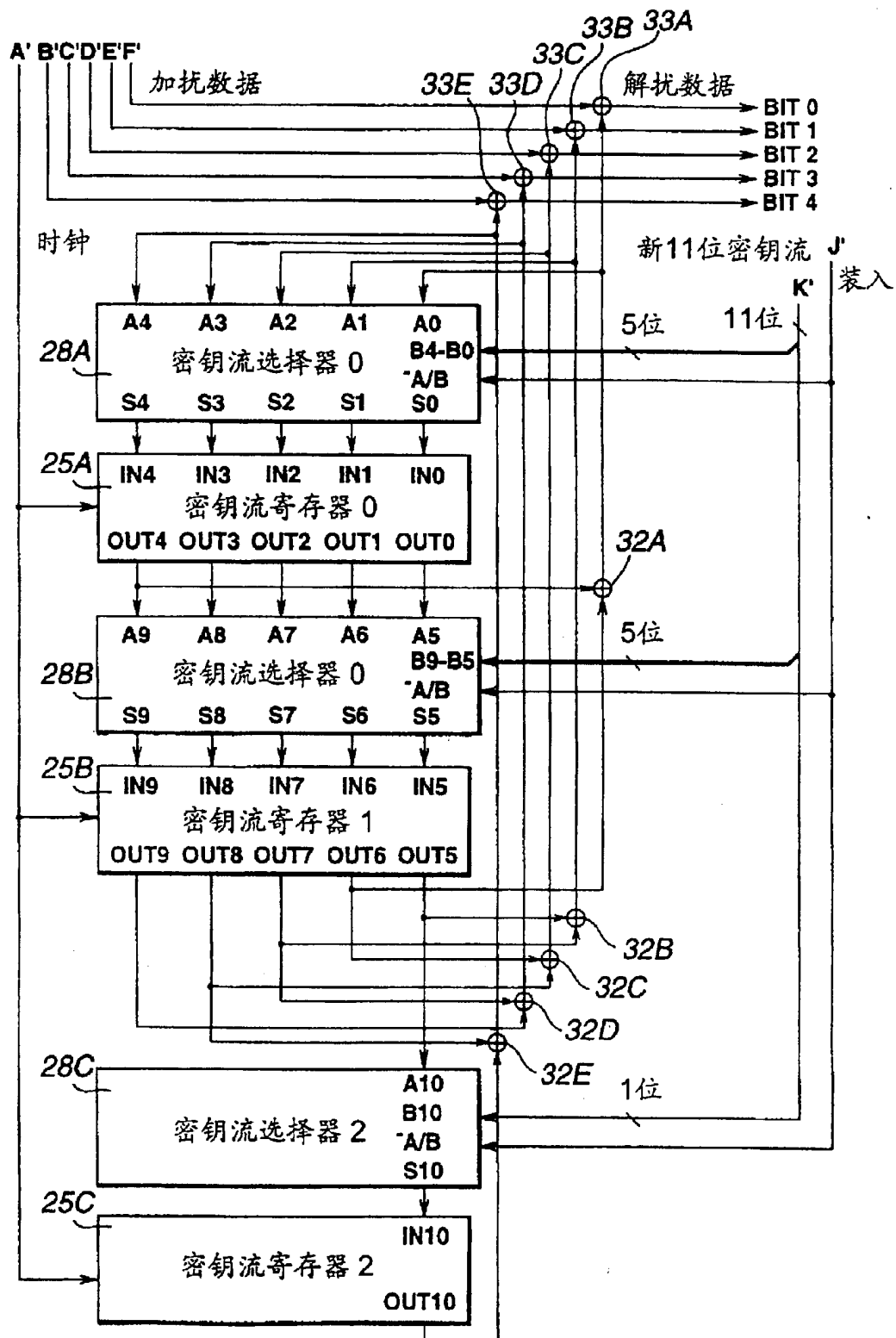


图 12

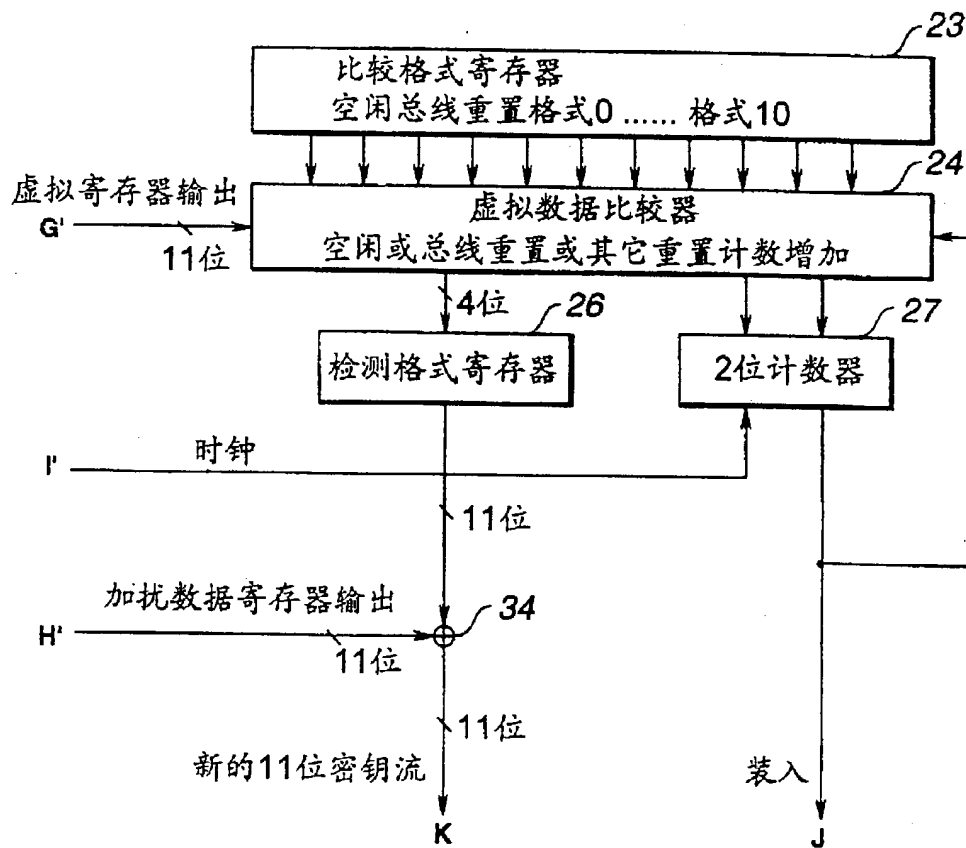


图 13

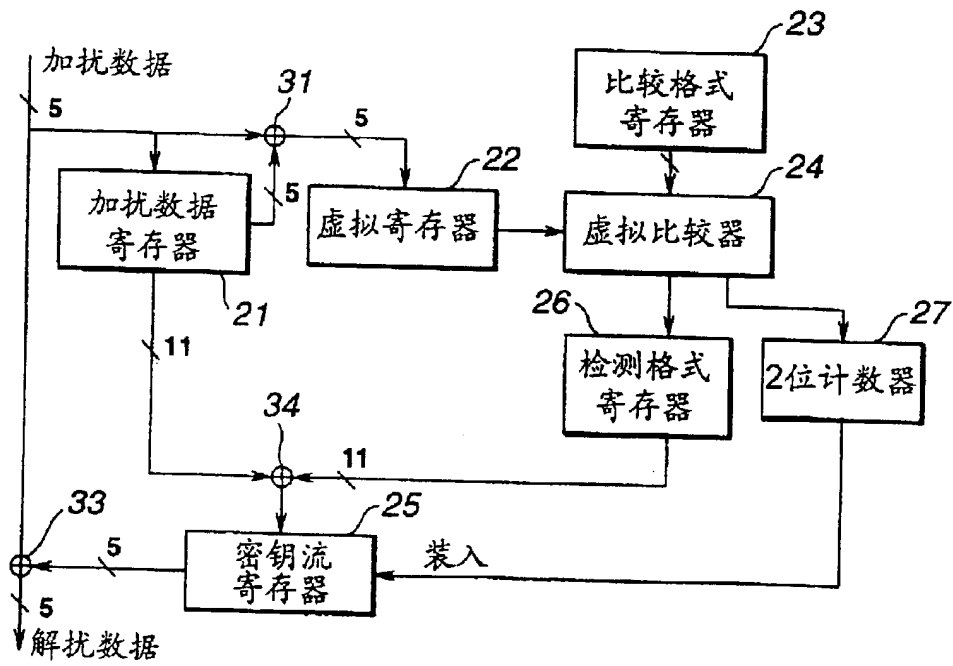


图 14

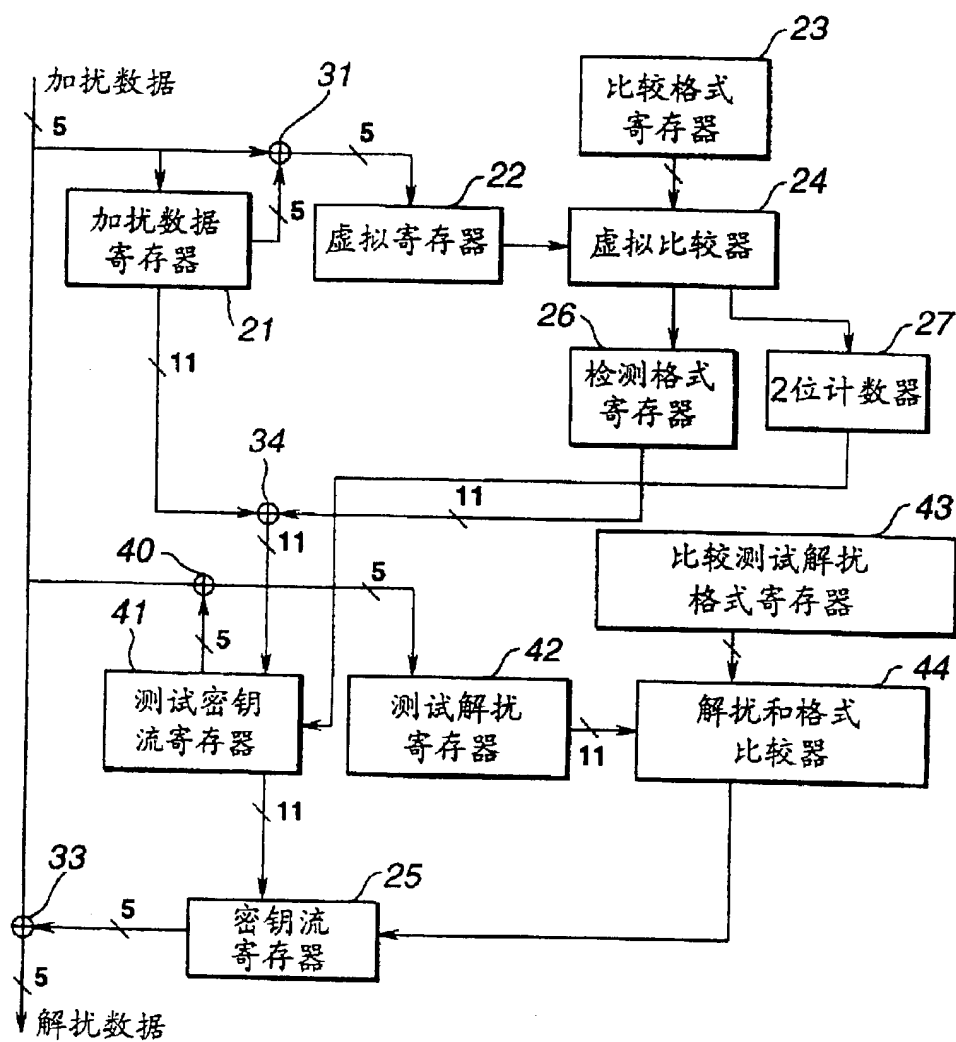


图 15